RUHR
UNIVERSITÄT
BOCHUM

RUB

Improving Anonymity in Public Group Communication Scenarios

zur Erlangung des akademischen Grades

Doktorin der Naturwissenschaften (Dr. rer. nat.)

von der Fakultät für Informatik
der Ruhr-Universität Bochum

genehmigte

Dissertation

von

Sarah Abdelwahab Kamel Fayed Gaballah

Tag der mündlichen Prüfung: 18.12.2024

Gutachterin: Prof. Dr. Karola Marky
Zweitgutachter: Prof. Dr. Max Mühlhäuser

ABSTRACT

Public group communication involves sharing information, ideas, and opinions within open groups that anyone can join and participate in. This form of communication is common across many domains, including *social networking* platforms, where users share posts and engage in discussions. It is also used in *crowdsourced* data-sharing platforms, where users collaboratively contribute to creating public datasets. In these contexts, *anonymity* can be crucial for individuals who share sensitive information and wish to prevent their identities from being linked to the information they provide. Many solutions that provide anonymity for public group communication have been proposed, yet they have limitations: solutions that offer strong anonymity are often inefficient, while solutions that prioritize efficiency provide weaker anonymity and are vulnerable to several attacks.

This dissertation makes several contributions to the field of anonymity in public group communication considering different scenarios. The first contribution is 2PPS, a protocol developed for social networking that allows users to publish messages and subscribe to content anonymously. The second contribution is Anonify, a protocol designed for crowdsourced data sharing, specifically for medical data donation. It allows individuals to share their data anonymously, ensuring their identity cannot be linked to their data through communication metadata (e.g., IP addresses or timestamps) or identifiable features in the shared data (e.g., demographic information or medical conditions). Both 2PPS and Anonify ensure efficiency and scalability while providing strong security and anonymity guarantees against powerful adversaries.

The third contribution of the dissertation is a comprehensive study of intersection attacks, a powerful traffic analysis threat that deanonymizes users by exploiting changes in anonymity sets over time. We present two variants of these attacks targeting anonymous public group communication and evaluate their effectiveness under realistic user communication patterns. Our findings indicate that these attacks are effective even when traditional countermeasures, such as cover

traffic or random delays, are employed. To protect against these attacks, the fourth contribution proposes a novel and efficient mitigation approach that forms anonymity sets with indistinguishable user behaviors, making it difficult for adversaries to identify individuals, even with prolonged monitoring.

For anonymity protocols to be effective, a large user base is crucial. The fifth contribution of this dissertation examines human factors, focusing on perceptions of anonymity in the context of medical data donation. Since this scenario involves sharing sensitive data without immediate benefits, understanding people's perceptions is especially important. Using semi-structured interviews and a drawing exercise, we captured expectations, wishes, and misconceptions. Based on our findings, we provide recommendations for designing user-centered, privacy-preserving medical data donation systems, ultimately increasing the number of participating users and, consequently, the size of anonymity sets.

ZUSAMMENFASSUNG

Öffentliche Gruppenkommunikation umfasst das Teilen von Informationen, Ideen und Meinungen innerhalb offen zugänglicher Gruppen, denen praktisch jeder beitreten kann. Diese Kommunikationsform ist in vielen Bereichen verbreitet, einschließlich sozialer Netzwerke, auf denen Benutzer:innen Beiträge teilen und sich an Diskussionen beteiligen. Weiterhin wird sie in sogenannten "Data-Sharing-Plattformen" (en: Crowdsourcing) verwendet, wo Benutzer:innen gemeinsam zur Erstellung öffentlicher Datensätze beitragen. In diesen Kontexten kann Anonymität entscheidend für Benutzer:innen sein, die sensible Informationen teilen und verhindern möchten, dass ihre Identität mit den von ihnen bereitgestellten Informationen verknüpft wird. Es wurden bereits zahlreiche Lösungen für Anonymität in öffentlicher Gruppenkommunikation vorgeschlagen. Allerdings haben diese Lösungen Einschränkungen: Einige ermöglichen zwar starke Anonymität, sind jedoch wenig effizient, während andere effiziente Kommunikation ermöglichen, aber eine schwächere Anonymität bieten und anfällig für verschiedene Angriffe sind.

Diese Dissertation leistet mehrere Beiträge im Bereich der Anonymität in der öffentlichen Gruppenkommunikation unter Berücksichtigung unterschiedlicher Szenarien. Der erste Beitrag ist 2PPS, ein Protokoll, das für soziale Netzwerke entwickelt wurde und es Benutzer:innen ermöglicht, Nachrichten anonym zu veröffentlichen und Inhalte zu abonnieren. Der zweite Beitrag ist Anonify, ein Protokoll, das für Crowdsourcing, speziell für das Spenden medizinischer Daten, konzipiert wurde. Es ermöglicht einzelnen Benutzer:innen ihre Daten anonym zu teilen und sicherzustellen, dass ihre Identität nicht mit den geteilten Daten verknüpft werden kann, was beispielsweise über Kommunikationsmetadaten (z. B. IP-Adressen oder Zeitstempel) oder identifizierbare Merkmale (z. B. demografische Informationen oder medizinische Diagnosen) möglich wäre. Sowohl 2PPS als auch Anonify gewährleisten Effizienz und Skalierbarkeit, während sie gleichzeitig starke Garantien für die Sicherheit und Anonymität gegen mächtige Angreifer bieten.

Der dritte Beitrag der Dissertation ist eine umfassende Studie über "Intersection Attacks", ein Deanonymisierungsangriff, welcher durch Netzwerkverkehrsanalysen Benutzer:innen de-anonymisiert, indem stattfindende Änderungen in den "Anonymity Sets" ausgenutzt werden. Die Dissertation präsentiert zwei Varianten dieses Angriffs, die auf anonyme öffentliche Gruppenkommunikation abzielen, und bewertet deren Wirksamkeit unter realistischen Kommunikationsmustern. Unsere Erkenntnisse zeigen, dass "Intersection Attacks" selbst dann effektiv sind, wenn traditionelle Gegenmaßnahmen wie "Cover Traffic" oder zufällige Verzögerungen (en: "random delays") eingesetzt werden. Um sich gegen diese Angriffe zu schützen, schlägt der vierte Beitrag einen neuartigen und effizienten Ansatz vor, der "Anonymity Sets" mit nicht unterscheidbaren Nutzendenverhalten bildet, was es Angreifern selbst bei längerer Überwachung erschwert einzelne Benutzer:innen zu identifizieren.

Damit Anonymitätsprotokolle wirksam sind, ist eine große Nutzendenbasis entscheidend. Der fünfte Beitrag dieser Dissertation untersucht menschliche Faktoren, wobei der Fokus auf den Wahrnehmungen von Anonymität im Kontext des Spendens medizinischer Daten liegt. Da dieses Szenario das Teilen sensibler Daten ohne unmittelbare Vorteile umfasst, ist das Verständnis der Wahrnehmungen der Benutzer:innen besonders wichtig. Durch semi-strukturierte Interviews und eine Zeichenübung haben wir Erwartungen, Wünsche und Missverständnisse erfasst. Basierend auf den Ergebnissen formulieren wir Empfehlungen für die Gestaltung nutzendenzentrierter, datenschutzfreundlicher Systeme zur Spende medizinischer Daten, um letztlich die Anzahl der teilnehmenden Benutzer:innen und folglich die Größe der "Anonymity Sets" zu erhöhen.

ACKNOWLEDGMENTS

This dissertation journey has been filled with challenges and successes, and it would not have been possible without the support, guidance, and encouragement of many incredible individuals.

First, I would like to express my heartfelt gratitude to Max Mühlhäuser for his guidance, trust in my abilities, and unwavering support, which gave me the confidence to tackle challenges and stay motivated throughout my PhD journey. I also owe special thanks to Karola Marky for her guidance, valuable insights that greatly improved my work, and for introducing me to the field of human-computer interaction.

Additionally, I am deeply thankful to all of my collaborators, especially Lamya Abdullah and Ephraim Zimmer from TU Darmstadt, with whom I closely collaborated throughout my PhD. I am also grateful to Christoph Coijanovic from KIT for the discussions that deepened my understanding of anonymous communication. I want to extend my thanks to my students, especially Thanh Hoang Long Nguyen and Minh Tung Tran, for their support.

Moreover, I would like to thank the TK family (TU Darmstadt), especially my colleagues from SPIN. I would also like to extend my thanks to my colleagues from the digisoul lab at RUB.

Finally, and most importantly, I would like to express my deepest gratitude to my parents, whose unwavering love, support, and encouragement have been the foundation of my success. Their sacrifices and belief in me have inspired me to keep pushing forward, and I am forever grateful to them. I also want to thank my siblings (Hagar, Mohammed, Ibrahim, and Yousef) for their constant support and understanding throughout this journey, always cheering me on. A big thank you to my friends as well, for their encouragement and for being there through both the highs and lows.

CONTENTS

LIST OF TABLES

Part I

SYNOPSIS

1 INTRODUCTION

1.1 MOTIVATION & PROBLEM STATEMENT

The internet has revolutionized the way humans communicate and share information: It has become an integral part of our lives by making connections faster and easier, enabling a wide range of applications like instant messaging, video calls, or social media. However, the internet has also opened a unique door to serious privacy risks, which have been demonstrated by numerous scandals over the past years. For example, the "Cambridge Analytica Scandal" revealed that a political consulting firm had harvested data from over 50 million Facebook users without their consent to influence voter behavior in elections [9]. Additionally, the "Snowden Revelations" uncovered extensive government surveillance programs, such as PRISM, which collected user data from major tech companies without users' knowledge [45]. These threats highlight the need to protect privacy in the online world.

Definition 1.1: Privacy

Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others [74, p. 7].

Privacy refers to the control individuals have over the information they share and how that information is collected, stored, used, accessed, and distributed (see also Definition 1.1). In certain scenarios, privacy protection should be extended to allow individuals to participate in online activities *without* revealing their true identities. For instance, this can be important for those who wish to share personal experiences related to mental health but are concerned about stigma or discrimination [8]. This can also be crucial for whistleblowers reporting unethical activities who may fear retaliation, job loss, or legal consequences [41].

Definition 1.2: Anonymity

Anonymity of a subject means that the subject is not identifiable within a set of subjects, the anonymity set [55, p. 9].

Concealing identities online is accomplished through *anonymity* (see Definition 1.2). Generally, the larger the anonymity set, the more difficult it is to identify an individual. Thus, the size of anonymity sets can be used to assess the degree of anonymization of subjects [55]. Anonymity can hide identifiable information about individuals within the data they share and in their communications.

Definition 1.3: Anonymous Communication

Anonymous communication refers to hiding network metadata and the relationships between communicating parties on the Internet [58, p. 1].

When used to safeguard communication, like in the case of anonymous web browsing, it is referred to as *anonymous communication* (see Definition 1.3). In this context, anonymity is maintained by obscuring communication/network metadata [58]. This metadata can reveal critical information, e.g., the identities of the communicating parties, as well as the timing and frequency of their interactions [25].

Definition 1.4: Data Anonymity

Data anonymity means that data cannot be linked to an identified or identifiable natural person, either because the data does not contain any personal identifiers or because it has been processed in such a way that the data subject is no longer identifiable (Recital 26, GDPR [51]).

When anonymity is applied to protect data, such as medical records, it is called *data anonymity* (see Definition 1.4). This form of anonymity is necessary for data that contains personally identifiable information (PII). Data anonymity is achieved by removing or altering PII, ensuring

that the data cannot be traced back to individuals [57]. The type of anonymity required—anonymous communication, data anonymity, or both—depends on the application scenario.

Anonymity has been extensively studied in both the domain of anonymous communication [48, 61] and data anonymity [10, 49], resulting in many solutions that achieve varying levels of anonymity. Strong anonymity solutions focus on ensuring that an individual's identity cannot be linked to their actions or data, even by powerful adversaries with substantial knowledge or control, such as nation-state actors. However, maintaining this level of anonymity while ensuring efficiency under realistic assumptions about users and their needs remains a challenge.

Definition 1.5: Group Communication

Group communication refers to the exchange of information among members of a group.

The complexity of providing anonymity increases further when communication involves multiple parties interacting with each other (i.e., *group communication*, see Definition 1.5). In this case, the identities of multiple senders and/or receivers within the same communication may need to be concealed [20, 21]. Achieving anonymity in group communication depends on the relationships among participants and their levels of trust. In *public* or *open groups*, where anyone can join and access shared content, maintaining anonymity is generally more difficult. This difficulty arises from the public nature of the shared content and the lack of trust among group members, who often have no prior personal connections and rely solely on mutual interests for communication [14].

This dissertation focuses on enhancing anonymity in public group communication. Specifically, it provides contributions related to two distinct group communication scenarios: 1) social networking and 2) crowdsourced data collection and sharing. In the following sections, we discuss these application scenarios, the importance of achieving anonymity within them, and the limitations of existing solutions in the literature.

1.1.1 *Social Networking*

In social networking platforms, users can connect, communicate, and share content with others, even without having any personal connection or established trust [50]. Anonymity in social networking can be vital for protecting free speech, particularly in environments where individuals may face censorship, persecution, or social stigma for expressing their views [54].

All well-known social networking platforms, such as Facebook [1] and Twitter [2] (rebranded as "X"), operate on centralized architectures [53], which allow these platforms to gather extensive information about their users. They often sell or disclose the data they collect to third parties, including governments. For instance, Facebook revealed that it provided data for 88% of the requests made by the U.S. government [71]. Such data disclosures can jeopardize the privacy of many users, including human rights activists or political dissidents.

Even if users share thoughts and opinions on social networks without including any personally identifiable information and use pseudonyms instead of their real names, their content can still be traced back to them through communication metadata, like IP addresses, message sizes, or timestamps [25]. The importance of communication metadata in identifying individuals is emphasized by statements like "We kill people based on metadata", made by former NSA Director Michael Hayden, and "Metadata absolutely tells you everything about somebody's life. If you have enough metadata, you don't really need content", as stated by former NSA General Counsel Stewart Baker [47].

To conceal communication metadata in the context of social networking, some solutions have been developed; however, these solutions encounter challenges:

CHALLENGE 1: *Weak Protection*

> Some systems offer considerably weak anonymity levels by providing probabilistic guarantees, rather than deterministic, provably secure cryptographic guarantees. Examples of such systems include those proposed by Daubert et al [20] and Giakkoupis et al [33].

1 https://www.facebook.com/ last accessed 16 October 2024
2 https://x.com/ last accessed 16 October 2024

Further, various proposed systems (e.g., developed by Corrigan-Gibbs et al. [15] and Lin et al. [44]) are susceptible to traffic analysis attacks. One example of such attacks is the *intersection attacks*, powerful attacks that can de-anonymize users by exploiting the changes in anonymity sets, resulting from changes in the users participating in the system over time [6, 35].

CHALLENGE 2: *Efficiency Issues*

Many solutions demonstrate inefficiencies in terms of latency (e.g., Dissent [16] and Atom [40]), requiring users to wait a significant amount of time for messages to arrive. Additionally, some solutions introduce high bandwidth overhead (e.g., Riposte [15] and Blinder [2]), resulting from the need to send and/or receive numerous messages, including dummy messages, to protect against adversaries observing communication. Furthermore, some existing solutions do not scale well (e.g., Dissent [16] and Riffle [39]), making it difficult for them to support large user bases.

In summary, the current solutions do not tackle both challenges. Hence, there is a need to develop approaches that address both issues to enable strong and efficient anonymous social networking for users.

1.1.2 *Crowdsourced Data Collection & Sharing*

Online platforms and mobile apps for crowdsourced data sharing facilitate the gathering of various types of data from a wide user base [38]. The collected information can be used to create valuable datasets for scientific research and innovation. For example, the Corona Data Donation app [63] allowed individuals during the pandemic to voluntarily share their medical information with researchers at the Robert Koch Institute. The collected data helped inform public health strategies and enhance understanding of the pandemic. However, given the potential sensitivity of shared data in such contexts, which may include PII, ensuring anonymity can be crucial to encourage user participation [8, 73].

Solutions for enabling privacy-preserving data collection from users have been proposed; yet, these solutions face the following challenges that extend those detailed in the context of social networking:

CHALLENGE 1: *Weak Protection*

Most existing solutions for crowdsourced data sharing focus on anonymizing the shared data, without addressing the anonymization of communication metadata. However, since individuals can be de-anonymized using metadata (e.g., location data), communication metadata should also be protected to ensure strong and comprehensive anonymity.

Another limitation in existing solutions like k-anonymity [70], ℓ-diversity [46], and t-closeness [43] is that these solutions are designed for centralized settings where a single entity aggregates and anonymizes all users' data [10]. This setup requires users to place great trust in that entity, as it possesses knowledge of each user's data. Furthermore, this centralized setting introduces a single point of failure: if an adversary gains control of this central entity, the anonymity of all users might be compromised.

CHALLENGE 2: *Efficiency Issues*

Similar to the social networking scenario, many solutions for protecting users in the context of crowdsourced data sharing suffer from inefficiencies. Solutions based on Secure Multi-Party Computation (SMPC) [67, 68, 75] often struggle with computational overhead, increased communication complexity, and limited scalability [81]. Also, solutions utilizing Homomorphic Encryption (HE) [65, 66, 79] suffer from high computational requirements and slower processing speeds [3].

Furthermore, solutions relying on SMPC, HE, and Differential Privacy (DP) [23] generally support only specific types of statistical analyses, limiting researchers' ability to conduct a wide variety of studies [76].

CHALLENGE 3: *Limited Consideration of Human Factors*

Researchers often prioritize technical security over human factors regarding anonymity. While strong anonymity is crucial, neglecting human factors — particularly in a scenario like anonymous

crowdsourced data sharing — can render systems ineffective. User participation is vital in this context because anonymous crowdsourced data sharing systems need users more than users need them, unlike anonymous social networking services, where the platform itself often drives engagement. Therefore, overlooking human factors in these systems might lead to systems that fail to attract users. Although some studies [8, 72, 73] emphasize the importance of anonymity in users' willingness to share data, there has been limited exploration of how individuals understand anonymity and the misconceptions they may hold. Additionally, the impact of these misconceptions on their willingness to share their data remains underexamined.

Overall, these limitations highlight the need for more robust solutions that ensure both data and communication anonymity, preventing any linkability between users and their shared data without relying on a single entity. These solutions should maintain efficiency. Moreover, they should optimize data utility and provide datasets to researchers that can be adapted to diverse research needs and analyses.

In this dissertation, we make several contributions to improving anonymity in public group communication scenarios. Our contributions are specifically related to three main objectives that specifically consider the challenges motivated above:

Providing Strong Anonymity: We propose novel protocols for anonymous social networking and anonymous crowdsourced data sharing (to address the challenge "weak protection" mentioned in Section 1.1.1 and 1.1.2). These protocols offer provable security guarantees against strong adversaries, such as global passive adversaries who control significant parts of the network. They also minimize trust requirements by leveraging the anytrust model [78], a decentralized client/server network model where clients (i.e., users) trust that at least one server executing the protocols behaves honestly. Additionally, these protocols protect against traffic analysis attacks, with one protocol specifically developed to effectively mitigate intersection attacks.

Improving Efficiency & Scalability: Addressing the second challenge "efficiency issues" (see Section 1.1.1 and 1.1.2), we prioritize efficiency in all our proposed solutions, ensuring they do not introduce prohibitive latency or bandwidth overhead for users. Additionally, recognizing that anonymity is more effective with a larger user base [56] (as users can blend into a larger crowd), our solutions are designed to support a large number of users without compromising efficiency. Moreover, for anonymous crowdsourced data sharing, the proposed solution provides anonymized data in a way that allows researchers to utilize it in different studies.

Considering Human Factors: To tackle the third challenge outlined in Section 1.1.2, we explore individuals' perceptions of privacy and anonymity in the context of medical data donation. By understanding people's needs, perspectives, and misconceptions, anonymous medical data donation systems can be designed to more effectively align with the privacy and anonymity requirements of data donors.

The results of our contributions can be summarized as follows:

1.3.1 *Anonymity Protocol for Social Networking*

In Chapter P1, this dissertation introduces a novel decentralized communication protocol named *2PPS* [32]. The aim of 2PPS is to facilitate public group communication effectively, particularly in social networking scenarios, while guaranteeing provable anonymity for both senders (publishers) and recipients (subscribers). A formal proof is provided to demonstrate that the 2PPS protocol ensures anonymity, even against a powerful adversary capable of monitoring all communications, controlling some servers or users, and launching active attacks. 2PPS distributes the functionality of the anonymity system across multiple servers, ensuring protection as long as at least one server does not collude with the adversary. While the adversary can learn which messages are published (as messages are published publicly), they cannot determine who published which message. This protection is achieved through an enhanced secret-sharing method that uses Distributed Point Functions [15], which also safeguards against active interference. To protect the topics that users subscribe to and the messages they receive from the system, 2PPS employs a method based on Private Information Retrieval [13, 34].

The evaluation of the 2PPS protocol highlighted its superior performance in terms of latency and bandwidth overhead compared to other systems like Riposte [15], Pung [4], and Blinder [2], which provide cryptographic anonymity guarantees similar to 2PPS. Additionally, results show that 2PPS maintained low latency even as the number of users in the system increased.

This contribution is the result of a publication at the 40th International Symposium on Reliable Distributed Systems, in collaboration with Christoph Coijanovic, Thorsten Strufe, and Max Mühlhäuser.

1.3.2 *Anonymity Protocol for Medical Data Donation*

This dissertation presents *Anonify* [28] in Chapter P2, a decentralized anonymity protocol designed to guarantee strong protection for data donors without depending on a single entity. Anonify provides dual-level anonymity protection, addressing both anonymous communication and data anonymity. This means it conceals communication metadata while processing donated data in a way that prevents adversaries from linking it back to individual users. The protocol employs a secret-sharing-based method for anonymous writing, utilizing Distributed Point Functions, along with a broadcasting-based approach for anonymous data retrieval. To mitigate de-anonymization risks related to donated medical data, Anonify incorporates k-anonymity [70] and stratified sampling [52], all within decentralized settings. A security analysis is provided to show that Anonify effectively achieves anonymous communication and data anonymity against an adversary capable of observing all communication, controlling parts of the anonymity system, and possessing background information about users (e.g., age and gender).

The evaluation of Anonify demonstrates its efficiency in preserving privacy while maintaining data utility. The protocol was tested on a realistic medical dataset, and the anonymized data retained its usefulness for analysis. Several machine learning classifiers were applied to the anonymized dataset, yielding results closely aligned with those from the original, non-anonymized data.

The contribution of this work is the result of a publication at the 24th Privacy Enhancing Technologies Symposium, in collaboration with Lamya Abdullah, Mina Alishahi, Thanh Hoang Long Nguyen, Ephraim Zimmer, Max Mühlhäuser, and Karola Marky.

1.3.3 *Understanding the Effectiveness of Intersection Attacks*

This dissertation in Chapter P3 contributes an in-depth study of intersection attacks [30] (see the "weak protection" challenge in Section 1.1.1). This study focuses on the effectiveness of attacks within anonymous social networking, specifically in the context of microblogging—a widely used social networking application scenario. Microblog-

ging was chosen as the use case because it enables the collection of real-world datasets that accurately reflect realistic user communication behaviors. This allows for evaluating how factors like sending rates influence the success of intersection attacks. Further, the study examines the effectiveness of common mitigation techniques, such as cover traffic and delays, in reducing the impact of these attacks.

Two variants of intersection attacks targeting anonymous microblogging systems were introduced: 1) the user-pseudonym linking attack, which aims to de-anonymize users who publish messages under pseudonyms, and 2) the user-topic linking attack, which targets systems where users post messages directly to topics without using pseudonyms or attaching any other personal identifiers. The evaluation results demonstrated that intersection attacks, especially the user-pseudonym linking attack, are highly effective, even when mitigation techniques like cover traffic or delays are employed.

The contribution of this work is the result of a publication at the 27th Nordic Conference on Secure IT Systems, in collaboration with Lamya Abdullah, Minh Tung Tran, Ephraim Zimmer, and Max Mühlhäuser.

1.3.4 *Mitigating Intersection Attacks*

To improve anonymity protection, this dissertation proposes, in Chapter P4, a protocol to mitigate intersection attacks in anonymous public group communication [27], focusing on microblogging as a use case for the reasons mentioned above. This protocol prevents intersection attacks by grouping users into sets based on the similarities in their publishing behavior. For each set, it generates a communication schedule that users must follow, ensuring that the users within the set appear indistinguishable to an adversary observing the entire communication.

An evaluation of the protocol was conducted using real-world datasets to simulate realistic user publishing behavior. This assessment examined the impact of schedule design on bandwidth and latency. The findings indicate that scheduling can reduce bandwidth overhead for users. However, as expected, this reduction often comes at the cost of increased latency, particularly for less active users who publish fewer messages. To address this, the protocol allows for adjusting schedules

to optimize the inevitable trade-off between bandwidth overhead and latency based on the needs of users in each set.

The contribution of this work is the result of a publication at the 18th International Conference on Availability, Reliability, and Security, in collaboration with Thanh Hoang Long Nguyen, Lamya Abdullah, Ephraim Zimmer, and Max Mühlhäuser.

1.3.5 *Understanding Privacy & Anonymity Perceptions of Medical Data Donation Apps*

In Chapter P5, this dissertation presents a qualitative study on how individuals perceive privacy and anonymity in the context of medical data donation apps [31]. Due to the limited use of these apps, the study focuses on the perceptions and expectations of non-users—those who have never used these apps. Specifically, the study explores the speculative mental models of 24 participants through semi-structured interviews and a drawing exercise.

The findings of the study indicate that most participants struggled to articulate a detailed mental model of their expected data donation infrastructure. They expressed greater trust in data donation apps from research institutes rather than commercial entities. Participants wanted control over their shared data but feared the complexities this might entail. Concerns about data breaches, misuse, and discrimination were prevalent, yet their understanding of these risks was limited. They wanted strong privacy guarantees, emphasizing the need for anonymity; most would refuse to use apps that could link data to their identities or locations. However, they were often unaware of the sensitivity of the data collected (including metadata) and the methods to ensure protection. Comparing participants' mental models with two existing medical data donation apps revealed significant gaps that could hinder adoption. To create user-friendly apps that meet privacy and anonymity expectations, several key design recommendations are provided.

The contribution of this work is the result of a publication at the 25th Privacy Enhancing Technologies Symposium, in collaboration with Lamya Abdullah, Ephraim Zimmer, Sascha Fahl, Max Mühlhäuser, and Karola Marky.

The peer-reviewed publications listed below constitute the main chapters of this dissertation. The research presented in these publications was conducted in collaboration with students, colleagues, and other participants involved in the respective research projects.

Chapter P1: 2PPS – Publish/Subscribe with Provable Privacy
Sarah Abdelwahab Gaballah, Christoph Coijanovic, Thorsten Strufe, and Max Mühlhäuser. 2021. 2PPS - Publish/Subscribe with Provable Privacy. In 40th International Symposium on Reliable Distributed Systems (SRDS), pp. 198-209. IEEE, 2021.

Chapter P2: Anonify: Decentralized Dual-level Anonymity for Medical Data Donation
Sarah Abdelwahab Gaballah, Lamya Abdullah, Mina Alishahi, Thanh Hoang Long Nguyen, Ephraim Zimmer, Max Mühlhäuser, and Karola Marky. "Anonify: Decentralized Dual-level Anonymity for Medical Data Donation." Proceedings on Privacy Enhancing Technologies 3 (2024): 94-108.

Chapter P3: On the Effectiveness of Intersection Attacks in Anonymous Microblogging
Sarah Abdelwahab Gaballah, Lamya Abdullah, Minh Tung Tran, Ephraim Zimmer, and Max Mühlhäuser. "On the Effectiveness of Intersection Attacks in Anonymous Microblogging." In 27th Nordic Conference on Secure IT Systems (NordSec), pp. 3-19. Springer, 2022.

Chapter P4: Mitigating Intersection Attacks in Anonymous Microblogging
Sarah Abdelwahab Gaballah, Thanh Hoang Long Nguyen, Lamya Abdullah, Ephraim Zimmer, and Max Mühlhäuser. "Mitigating Intersection Attacks in Anonymous Microblogging." In 18th International Conference on Availability, Reliability and Security (ARES), pp. 1-11. ACM, 2023.

Chapter P5: "It's Not My Data Anymore": Exploring Non-Users' Privacy Perceptions of Medical Data Donation Apps
Sarah Abdelwahab Gaballah, Lamya Abdullah, Ephraim Zimmer, Sascha Fahl, Max Mühlhäuser, and Karola Marky. "'"It's Not My Data Anymore": Exploring Non-Users' Privacy Perceptions of Medical Data Donation Apps." Proceedings on Privacy Enhancing Technologies 1 (2025): 654–670.

This dissertation is structured into two parts. **Part i** includes the synopsis and consists of this chapter (Introduction), along with:

CHAPTER 2 details the fundamentals of this dissertation by providing an in-depth overview of anonymous communication and data anonymity.

CHAPTER 3 concludes the dissertation by summarizing the main findings, explaining how they can be extended, discussing their implications, and outlining potential directions for future work.

Part ii includes the publications that form the cumulative part of this dissertation, presenting each publication in its original form:

CHAPTER P1 introduces 2PPS, an anonymity protocol designed for social networking.

CHAPTER P2 presents Anonify, an anonymity protocol designed for medical data donation.

CHAPTER P3 investigates the effectiveness of intersection attacks in de-anonymizing social network users.

CHAPTER P4 proposes an efficient mitigation protocol to protect against intersection attacks.

CHAPTER P5 explores perceptions of privacy and anonymity in the context of medical data donation.

2

BACKGROUND

This chapter provides an overview of anonymous communication (Section 2.1) and data anonymity (Section 2.2).

2.1 ANONYMOUS COMMUNICATION

Anonymous communication allows individuals to exchange information without revealing their identities or any traceable details. This is typically achieved by encrypting messages to hide the exchanged content and by concealing communication metadata.

Communication metadata refers to the properties or details of the communication [25]. Examples of communication metadata are:

- the identities of senders and receivers (e.g., their email addresses or IP addresses)
- timestamps (e.g., when a message was sent or received)
- location data (e.g., the geographical position of a user when sending a message)
- network identifiers (e.g., MAC addresses or device IDs)
- the duration of communication (e.g., the length of a phone call or online session)
- the type of communication (e.g., email, voice call, or chat)

Collecting and analyzing communication metadata can introduce several risks, like tracking the users' online activities or revealing sensitive information about their relationships and behaviors. To mitigate these risks, anonymous communication systems are designed to conceal communication metadata.

Our contributions in Chapters P1, P2, and P4 focus on protocols designed to protect communication metadata. The contribution in

Chapter P3 examines how communication metadata, when combined with other information, can be used to de-anonymize users in real-world public group communication scenarios. Chapter P5 includes an investigation of people's awareness of different types of data, including metadata.

The rest of this section is organized as follows: privacy goals are discussed in Section 2.1.1, different types of adversaries are covered in Section 2.1.2, traffic analysis attacks are examined in Section 2.1.3, anonymous communication primitives are detailed in Section 2.1.4. Finally, latency and bandwidth overhead are addressed in Section 2.1.5.

2.1.1 *Privacy Goals*

There are different privacy goals that can be achieved by anonymous communication systems. These goals include [55]:

SENDER ANONYMITY aims to hide the sender of a message. For instance, this is needed when a whistleblower wants to report misconduct while hiding their identity.

RECEIVER ANONYMITY aims to hide the receiver of a message. This protection could be necessary for individuals who want to, for example, subscribe anonymously to content about sensitive topics, such as a stigmatized medical condition (e.g., HIV/AIDS), without disclosing their identity.

RELATIONSHIP ANONYMITY aims to hide who is communicating with whom. A situation that illustrates the need for this protection is when two activists communicate online to organize a protest and want to ensure their connection remains hidden from an oppressive regime.

In Chapters P1 and P2, we ensure both sender and receiver anonymity. Chapter P3 focuses on de-anonymizing sender anonymity, while the contribution in Chapter P4 aims to protect sender anonymity against intersection attacks.

2.1.2 *Different Types of Adversaries*

Anonymous communication systems differ in the types of adversaries they aim to defend against [25]. The following represents the common powerful adversarial models:

GLOBAL PASSIVE ADVERSARY : This adversary can eavesdrop on the entire network but does not interfere with or modify the traffic.

GLOBAL ACTIVE ADVERSARY : This adversary can monitor the whole network and perform active attacks, such as modifying traffic content and injecting arbitrary messages.

GLOBAL PASSIVE ADVERSARY WITH CORRUPTION : This adversary is capable of eavesdropping on the entire network while also compromising entities within the network (e.g., some relay nodes or users). These capabilities allow the adversary to conduct various attacks, leveraging insider knowledge alongside passive surveillance. For example, this adversary can carry out attacks that disrupt communication, thereby preventing users from sending or receiving messages.

GLOBAL ACTIVE ADVERSARY WITH CORRUPTION : This is the most comprehensive model, encompassing all the capabilities of the previously mentioned adversaries.

In Chapter P1, we focus on protection against a global active adversary with corruption. In Chapter P2, we defend against a global passive adversary with corruption. In Chapters P3 and P4, we consider a global passive adversary.

2.1.3 *Traffic Analysis Attacks*

Traffic analysis attacks are passive attacks where an adversary tries to match the sender of a message to its recipient by monitoring metadata, like the timing, size, or frequency of messages [18]. These attacks are challenging to detect because the adversary simply monitors and analyzes the traffic without interfering with or altering it. Furthermore, when these attacks are carried out over a long duration, they allow

adversaries to construct detailed profiles of communication patterns, making them more effective and harder to mitigate [6].

Below, we describe two popular examples of traffic analysis attacks: timing attacks [1] and intersection attacks [77].

2.1.3.1 *Timing Attacks*

Timing attacks target systems that rely on real-time, low-latency communication where the timing characteristics of transmitted messages remain largely unchanged [37]. Tor is a well-known example of such a system [22]. In this type of attack, an adversary observes the timing patterns of incoming and outgoing messages and correlates them to identify who is communicating with whom [36, 37].

Example 2.1: Timing Attacks

Alice and Bob use an anonymity system to communicate privately. This system promptly forwards all messages of the users to the designated recipients. An attacker, Eve, monitors the incoming and outgoing messages of the system. She notices that every time Alice sends a message to the system, Bob receives a message from the system shortly afterward, and vice versa. By correlating these timing patterns, Eve deduces that Alice and Bob are communicating with each other, thus compromising their anonymity.

To mitigate against these attacks, contributions in Chapters P1, P2, P3, and P4 employ round-based communication, where users send messages within specified time intervals, and all messages from that interval are published simultaneously once the interval ends.

2.1.3.2 *Intersection Attacks*

These attacks exploit changes in anonymity sets caused by variations in the group of users participating in an anonymity system over time [7, 35]. By monitoring communications, an adversary can intersect these anonymity sets to identify which senders and receivers are active si-

multaneously over time, thus revealing potential relationships between them. Intersection attacks can also be applied in public group communication scenarios, such as online social networking, to determine who is publishing specific content [77].

Intersection attacks can be executed in two ways: The first approach is deterministic, meaning that if the attack is successful, the adversary can identify a user with absolute certainty. For example, this could involve determining that a specific user is definitively communicating with another particular user or that a user is indeed the owner of a specific anonymous social media account. The second approach is probabilistic, known as a statistical disclosure attack [17]. This approach aims to identify users based on probability, i.e., estimating the likelihood that two users are communicating or that a specific user is the owner of a given account [35].

Example 2.2: Intersection Attacks

A corrupt mayor learns that a user on an anonymous social networking platform has been posting allegations about the mayor's corruption or illegal/immoral acts. To identify this anonymous user, the mayor pressures local internet service providers to track and compile a list of all users active on the platform during the specific times when the posts from the accused account were made. The mayor's office requests access to connection logs, which include timestamps of when users accessed the platform. Initially, each list contains hundreds of users, but as the mayor's team intersects these lists over time, they can narrow down the timeframe of the posts and identify unique user patterns. This shrinks the pool of potential suspects gradually. Eventually, they identify a single user whose posting times match those of the anonymous user.

Chapter P3 offers an in-depth analysis of these attacks, while Chapter P4 presents an effective mitigation solution for them.

2.1.4 *Anonymous Communication Primitives*

Several methods can be employed to achieve anonymous communication. The following are some of the most popular primitives. This dissertation leverages some of these primitives to propose solutions that offer stronger anonymity guarantees and improved efficiency, particularly in the context of public group communication.

2.1.4.1 *Proxy Servers*

Using a proxy server to forward messages is the simplest method for ensuring anonymous communication. In this approach, the sender transmits the message to the proxy, which then forwards it to the intended recipient, preventing the adversary from directly connecting the sender and receiver [60]. However, this method relies on trusting the proxy to maintain privacy. Using a single proxy poses a critical risk; if an adversary gains control of the proxy, they can associate all senders with their respective receivers.

2.1.4.2 *Onion Routing*

Onion routing [59] is a well-known anonymization method, with Tor being its most prominent implementation [22]. This method works by encrypting a message in multiple layers by the sender. The encrypted message is then transmitted through a series of nodes (relays) to obscure the link between the sender, the message, and the intended recipient. Each node decrypts one layer to reveal the next destination, knowing only its immediate predecessor and successor. The entry node (the first node in the path) knows the sender, while the exit node (the last node in the path) knows the receiver. To guarantee anonymity protection, at least one of the nodes along the message transmission path must be trustworthy. Although onion routing can offer relatively low latency for anonymous communication, it is susceptible to traffic analysis attacks, such as timing attacks [37, 69] (see Section 2.1.3).

2.1.4.3 *Mix Networks*

Mix networks [12] provide anonymity protection to users by routing messages through several nodes called mix nodes or mixes. At each mix node, messages from multiple senders are collected and shuffled. This process conceals who is communicating with whom, especially against a global adversary capable of monitoring the entire communication within the network. Mix nodes can introduce random delays before forwarding messages to the next hop, meaning the forwarding is delayed by a random amount of time. These delays make it more difficult for adversaries to correlate incoming and outgoing communications, hence reducing the risk of traffic analysis. However, the introduction of random delays can lead to high latency, which is unsuitable for scenarios where users require instant communication.

2.1.4.4 *Dining Cryptographers Networks*

Dining Cryptographers Networks (DC-Nets) [11] represent a popular method for achieving anonymous communication. This method allows a user to send a message anonymously to a group of users. Typically, communication occurs in rounds, with one message transmitted in each round. Each user broadcasts a secret share to all others, and one share contains the message to be sent. All shared secrets appear random, ensuring only the sender knows their share contains the message. To uncover the sent message, users combine their secret shares with those received from others. If multiple users send messages in the same round, it can lead to collisions, hindering message retrieval [16]. Another limitation of DC-Nets is the high bandwidth overhead incurred because they rely on a broadcasting scheme.

2.1.4.5 *Private Information Retrieval*

Private Information Retrieval (PIR) [13] is a cryptographic method that allows a user to retrieve a specific data point from a database without revealing which data is being accessed. This can be achieved by distributing the database across multiple servers, where at least one of these servers should be honest. To obtain the desired information, the user sends a query to each server, with each query designed to conceal the user's interest. By combining the responses from all servers,

the user can uncover the desired information without disclosing which specific data was requested. This dissertation provides a PIR-based solution in Chapter P1.

2.1.4.6 *Distributed Point Functions*

The Distributed Point Functions (DPF)-based method [15] is a secret-sharing technique. It allows users to write messages anonymously to a database distributed across multiple servers. It ensures that no individual server can determine what each user has written.

To write a message m, a user generates a set of shares:

$$\{f_1, f_2, \ldots, f_N\} = \text{GenDPF}(m, x)$$

N is the number of servers and x is a randomly chosen index in the database. DPF compresses these shares to minimize bandwidth overhead. The user submits one share to each server., which adds it to its instance of the distributed database. Servers decompress these shares before adding them to their database instances. After receiving shares from multiple users, the servers collaborate to reveal the written messages by combining their database instances. DPF is used in solutions provided in Chapters P1 and P2.

2.1.4.7 *Broadcasting*

Broadcasting can be used to protect receiver anonymity by delivering a message to a large group of potential recipients, while only the intended recipient can interpret or gain value from the message. Since everyone in the group receives the same message, it becomes difficult for an adversary to determine who the message was intended for. Broadcasting is used in Chapter P2.

2.1.4.8 *Cover Traffic*

Cover traffic improves anonymity by adding noise to communication patterns, making it challenging to differentiate between genuine and dummy messages [58]. For this, users can create fake messages

containing random content. These messages blend with real communications, hence increasing the size of anonymity sets. The cover traffic can be transmitted either randomly or systematically (e.g., at regular intervals). Even though cover traffic has the benefits detailed above, it comes at the expense of increased bandwidth usage. Cover traffic is considered in Chapters P1, P3, and P4.

2.1.5 *Latency & Bandwidth Overhead*

After presenting several existing primitives for facilitating anonymous communication, we will discuss two key implications: the increased latency and bandwidth overhead associated with such privacy protection. Compared to traditional communication systems without privacy protections, anonymous communication systems often come at the cost of higher bandwidth and latency [19].

BANDWIDTH OVERHEAD refers to the additional packets that need to be transmitted to preserve user anonymity, resulting in increased overall data traffic within the network. This can put a strain on network resources, particularly in environments with limited bandwidth availability.

LATENCY OVERHEAD is the increased time it takes for a message to be delivered to a recipient. For example, this delay can result from the extra hops that anonymized packets must make to obscure their origin and destination. Each additional hop can add latency, which may not be acceptable for applications requiring real-time communication, e.g., Voice over IP (VoIP).

Completely avoiding bandwidth overhead or latency in anonymity systems is not possible [19]. However, some solutions incur considerably higher bandwidth overhead (e.g., DC-Nets and broadcasting) or greater latency (e.g., mix networks) compared to others. In Chapters P1, P2, and P4, we present contributions aimed at maintaining low latency and bandwidth overhead while guaranteeing strong anonymity.

Data anonymization is the process that involves the removal or alteration of personally identifiable information (PII) from datasets in such a way that the identities of individuals cannot be linked back to specific data points [57].

The remainder of this section is structured as follows: types of attributes in Section 2.2.1, privacy threats in Section 2.2.2, data anonymization techniques in Section 2.2.3, and data utility in Section 2.2.4.

2.2.1 *Types of Attributes*

Datasets are typically represented as relational (tabular) data, which is organized into columns (attributes) and rows (records). Attributes can be classified into four categories [49]:

DIRECT IDENTIFIERS (IDS) contain information that explicitly identifies the owners of records, such as names or social security numbers.

QUASI-IDENTIFIERS (QIDS) are attributes, like age, occupation, gender, and zip code, that cannot uniquely identify an individual on their own. However, when some of these attributes are combined, they could potentially lead to identification.

SENSITIVE ATTRIBUTES (SAS) are personal attributes of a private nature that should remain confidential (e.g., salaries or medical conditions).

NON-SENSITIVE ATTRIBUTES refer to all attributes that do not fit into any of the other three categories.

2.2.2 *Privacy Threats*

To protect individuals' privacy in datasets, direct identifiers must be removed. However, Sweeney [70] demonstrated that this step alone is insufficient to mitigate re-identification risks, such as *identity disclosure*

and *attribute disclosure* attacks. These attacks are relevant to this dissertation because they are addressed in the contributions presented in Chapter P2. Below, we discuss these attacks in more detail.

2.2.2.1 *Identity Disclosure*

Identity disclosure occurs when an individual can be linked to their records in a published dataset [10]. This risk arises when anonymized or aggregated data still contains information that enables an adversary to re-identify individuals. While direct identifiers are typically removed from published datasets to safeguard privacy, an adversary can link data by matching QIDs. To succeed, an adversary must have prior knowledge of the victim's QIDs, which can be obtained from public records, social media, or other online sources. Unique QIDs allow an adversary to narrow down potential matches, and even a small number of these attributes can identify individuals, especially in smaller populations or among rare characteristics.

Example 2.3: Identity Disclosure Attacks

An adversary has access to a dataset, such as the one presented in Table 2.1, and knows that Alice has a record in it. Additionally, the adversary is aware of Alice's QIDs: her age is 31, her gender is female, her occupation is scientist, her race is white, and her zip code is 12345. With this information, the adversary can identify from the dataset that Alice has cancer.

Age	Occupation	Zip Code	Gender	Race	Disease
29	Teacher	12345	Female	White	Diabetes
34	Engineer	12345	Male	Black	Hypertension
30	Doctor	67890	Female	Asian	Asthma
28	Nurse	67890	Male	Hispanic	Allergies
31	Scientist	12345	Female	White	Cancer
45	Lawyer	12345	Male	Asian	Heart Disease
50	Chef	67890	Female	Black	Obesity
38	Artist	12345	Male	Hispanic	Depression
26	Engineer	67890	Female	White	Anxiety

Table 2.1: Example of Identity Disclosure

2.2.2.2 *Attribute Disclosure*

Attribute disclosure occurs when an individual is linked to specific information about their SAs, which can be either the exact value of SA or an estimate [49]. While attribute disclosure often results from identity disclosure, it can also occur independently [49]. These attacks can happen even in anonymized datasets, where each record shares the same QIDs with a group of other records, making it indistinguishable from others in the group based on those QIDs. Such attacks are particularly likely when there is a lack of diversity in SA values among records within these groups.

Example 2.4: Attribute Disclosure Attacks

In Table 2.2, group 1 consists of individuals who share the same sensitive attribute (hepatitis). If an adversary knows that Alice belongs to this group, they can directly infer that she has Hepatitis, despite the anonymization based on QIDs. In group 2, while individuals have different diagnoses (heart disease, high blood pressure, arrhythmia), all conditions are heart-related. Thus, an attacker could still deduce that anyone in this group likely has a heart-related issue.

Group	Age	Gender	Zip Code	Disease
1	30-35	Female	12345	Hepatitis
1	30-35	Female	12345	Hepatitis
1	30-35	Female	12345	Hepatitis
2	40-45	Male	54321	Heart Disease
2	40-45	Male	54321	High Blood Pressure
2	40-45	Male	54321	Arrhythmia

Table 2.2: Example of Attribute Disclosure

2.2.3 *Data Anonymization Techniques*

Data can be anonymized using various techniques, and here we highlight some of the most common methods. In Chapter P2, we compare our proposed work to the solutions outlined in this section. We also

present a solution that incorporates several methods, one of which is *k*-anonymity (see below). However, this solution applies *k*-anonymity in a more secure manner than the traditional approach discussed in this section.

2.2.3.1 *k-anonymity*

k-anonymity [70] ensures that at least *k* individuals share the same QID values. A dataset satisfies *k*-anonymity if every unique combination of QID values appears in at least *k* records, forming an equivalence class.

Example 2.5: *k*-anonymity

Table 2.3 illustrates 5-anonymity, as each unique combination of quasi-identifiers (Gender, Age, and Zip Code) exists in at least 5 records. Specifically, there are 5 records for the combination of (Gender: Male, Age: 30-35, Zip Code: 12345) and another 5 records for the combination of (Gender: Female, Age: 25-30, Zip Code: 12345), ensuring that individuals cannot be easily identified based on these QID values. In this example, age is generalized to prevent attackers from identifying individuals' records based on exact ages. By generalizing age, we ensure that all individuals within the same equivalence class are indistinguishable in terms of age.

ID	Gender	Age	Zip Code	Test Result
1	Male	30-35	12345	Positive
2	Male	30-35	12345	Negative
3	Male	30-35	12345	Positive
4	Male	30-35	12345	Negative
5	Male	30-35	12345	Positive
6	Female	25-30	12345	Negative
7	Female	25-30	12345	Negative
8	Female	25-30	12345	Negative
9	Female	25-30	12345	Positive
10	Female	25-30	12345	Positive

Table 2.3: Example of 5-anonymity

k-anonymity can ensure protection against identity disclosure (see Section 2.2.2.1). The value of k determines the level of privacy, where higher k values ensure greater privacy protection. However, raising the k value may necessitate more generalization, resulting in information loss and, consequently, reducing data utility. The effectiveness of k-anonymity also depends on the diversity of SA values within an equivalence class. If there is limited or no diversity in the values of SAs, k-anonymity cannot guarantee protection against attribute disclosure attacks.

2.2.3.2 *ℓ-diversity*

ℓ-diversity [46] addresses the limitations of k-anonymity by ensuring diversity among SAs. It requires that each equivalence class formed based on QIDs contains at least ℓ distinct SA values.

Example 2.6: ℓ-diversity

Table 2.4 satisfies 3-diversity because each equivalence class contains 3 distinct SA values.

ID	Gender	Age	Zip Code	Sensitive Attribute (SA)
1	Male	30-35	12345	Hypertension
2	Male	30-35	12345	Diabetes Type 2
3	Male	30-35	12345	Asthma
4	Male	30-35	12345	Hypertension
5	Male	30-35	12345	Diabetes Type 2
6	Male	30-35	12345	Asthma
7	Female	40-45	54321	Heart Disease
8	Female	40-45	54321	Osteoporosis
9	Female	40-45	54321	Arthritis
10	Female	40-45	54321	Heart Disease
11	Female	40-45	54321	Osteoporosis
12	Female	40-45	54321	Arthritis

Table 2.4: Example of 3-diversity

Although ℓ-diversity aims to reduce the risk of attribute disclosure attacks, it does not eliminate this risk entirely. For instance, the equivalence class shown in Table 2.5 satisfies 3-diversity by including three

SA values: coronary heart disease, arrhythmia, and valve disease. However, an adversary could still deduce that any individual with a record in this class likely has a heart-related condition, as all three diseases are related to heart health.

ID	Gender	Age	Zip Code	Disease
1	Male	60-65	12345	Coronary Heart Disease
2	Male	60-65	12345	Arrhythmia
3	Male	60-65	12345	Valve Disease
4	Male	60-65	12345	Coronary Heart Disease
5	Male	60-65	12345	Arrhythmia
6	Male	60-65	12345	Valve Disease

Table 2.5: Example of Attribute Disclosure in a 3-diverse Class

Another limitation of ℓ-diversity is its ineffectiveness with skewed data distributions [43]. For example, consider a scenario where an adversary is aware that 95% of patients in a dataset have the flu, while only 5% have HIV. If an equivalence class within this dataset contains 50% of patients with the flu and 50% with HIV, it does not adequately protect privacy, even though it is 2-diverse. This is because the information in this class enables an attacker to deduce that an individual within this class is significantly more likely to have HIV compared to a randomly selected individual from the entire dataset.

2.2.3.3 *t-closeness*

t-closeness [43] addresses attribute disclosure and mitigates risks related to skewed data distributions. An equivalence class is said to have t-closeness if the distance between the distribution of SAs in that class and the distribution of SAs in the overall dataset is at most t. A dataset achieves t-closeness if this condition is satisfied for all its equivalence classes. A smaller value of t indicates stronger privacy protection. Typically, the Earth Mover Distance (EMD) function [64] is employed to measure the proximity between two distributions of sensitive values.

An example of t-closeness can be illustrated with a dataset in which 60% of individuals have the flu as their disease. To satisfy t-closeness, each equivalence class within the dataset must closely mirror this distribution, meaning that around 60% of the members in any class should also have the flu. Although an attacker could deduce that any

individual in the class has the flu with a probability of about 60%, the class remains protected under t-closeness because this likelihood is consistent with what can already be inferred from the overall dataset.

Despite its strengths, t-closeness has limitations. One major issue is its inability to accommodate different protection levels for various SAs [26], even though some values may be more sensitive than others. For instance, having the flu as an SA is less sensitive than having HIV, yet the parameter t is applied uniformly in both cases. Additionally, t-closeness can significantly reduce the utility of the data, as it requires the distribution of sensitive values to remain consistent across all equivalence classes. Moreover, the EMD function may not effectively prevent attribute disclosure attacks on numerical SAs [42].

2.2.3.4 *Differential Privacy*

Differential Privacy (DP) [23] ensures that the inclusion or exclusion of an individual's record in a dataset has minimal impact on the results of any analysis performed. This is achieved by adding noise to the data. There are two approaches for adding noise to satisfy DP [49]:

GLOBAL DP: The data aggregator, responsible for collecting and publishing the data, adds noise to the outputs generated from queries made by individuals interested in the aggregated dataset.

LOCAL DP: Noise is typically added to each data point (i.e., record) by the record owner before the record is shared.

In local DP, each record is distorted independently without considering the overall dataset, which may result in greater distortion of records than necessary. In contrast, global DP can take into account the characteristics of the dataset, often leading to more accurate results. However, with global DP, record owners must trust the data aggregator to preserve their privacy.

Unlike k-anonymity, ℓ-diversity, and t-closeness, which result in publishing the entire dataset, DP was initially designed for interactive query settings. In these settings, individuals interested in the data do not have direct access to the complete dataset; instead, they send queries to an aggregator that manages the dataset. It was later shown that DP can also accommodate non-interactive contexts, where data is

preprocessed using the DP mechanism and then released to the public for independent statistical analysis [24]. In this case, anyone can utilize the data to compute answers to various queries without needing to interact with the aggregator.

2.2.4 *Data Utility*

Data anonymization techniques modify data to meet privacy requirements, often at the expense of data utility. Generally, there is an inherent trade-off between data utility and data privacy [10]. Therefore, these techniques must find a balance between achieving the desired level of data protection and maintaining the usefulness of the data [49]. Furthermore, when evaluating a specific data anonymization technique, it is essential to assess not only the privacy it provides but also the utility of the anonymized data. Various metrics, e.g., the discernibility metric (DM) [5] and the normalized certainty penalty (NCP) [80], can be employed to measure data utility.

3

SUMMARY AND FUTURE WORK

Anonymity in online communication has been extensively studied for many years, but achieving strong anonymity while ensuring efficiency remains a challenge. This is particularly pronounced when adversaries possess extensive capabilities, such as monitoring or controlling large parts of the network or launching active attacks. Achieving anonymity becomes more difficult in scenarios involving public group communication, where the open nature of the published content and the potential lack of trust between the communicating parties add to the complexity.

This dissertation makes several contributions to the field of anonymous public group communication by enhancing anonymity, efficiency, and scalability while also considering human factors. Our work mainly addresses two key application scenarios which are: 1) social networking, and 2) crowdsourced data sharing (specifically, the donation of medical data).

This chapter first outlines the main contributions and findings of the dissertation (see Section 3.1). Then, the implications of the results and potential extensions to other scenarios and settings are discussed in Section 3.2. Finally, Section 3.3 suggests directions for future research.

3.1.1 *Strong & Efficient Anonymity for Social Networking*

Chapter P1 introduced 2PPS (Twice-Private Publish-Subscribe), a decentralized pub/sub protocol designed to provide strong, provable privacy protection for users in the context of social networking, enabling them to publish messages and subscribe to content of interest anonymously. For anonymous publishing, 2PPS uses an approach based on Distributed Point Function-based secret sharing, and for anonymous subscribing, it employs a method based on Private Information Retrieval. Additionally, it implements several security and privacy measures to defend against traffic analysis and active attacks. According to our evaluation, 2PPS is efficient in terms of latency and bandwidth overhead, even as the number of users in the system increases.

3.1.2 *Strong & Efficient Anonymity for Medical Data Donation*

In Chapter P2, we presented Anonify, our decentralized anonymity protocol designed to provide strong protection for users when they donate their medical data. It ensures that an adversary cannot deanonymize users by exploiting communication metadata (e.g., IP addresses) or through identifiable characteristics (e.g., demographic information) within the data that users share. This means it provides dual levels of protection for data donors: anonymous communication and data anonymity. This is achieved by applying a method based on Distributed Point Functions for anonymous data submission and a broadcasting-based approach for anonymous data retrieval. To ensure data anonymity, Anonify employs two techniques: k-anonymity and stratified sampling, in a decentralized manner, without trusting a single entity. Our evaluation demonstrates that Anonify effectively balances anonymity preservation with data utility. Additionally, the performance of machine learning algorithms on datasets anonymized by our protocol shows high accuracy and precision.

3.1.3 *Analysis of Intersection Attacks in Anonymous Microblogging*

In Chapter P3, we investigated the effectiveness of intersection attacks in a popular anonymous public group communication scenario—anonymous microblogging, which is a use case of anonymous social networking. Our analysis considered realistic user communication settings by utilizing datasets from Twitter [1] and Reddit [2]. We developed two variants of intersection attacks: one targeting anonymous microblogging systems that rely on pseudonym-based messaging patterns, and the other targeting systems that rely on topic-based messaging patterns. Our study shows that these attacks are effective regardless of whether users post messages under pseudonyms or publish them to topics without attaching pseudonyms. We found that users with high message-sending rates are particularly vulnerable, especially when the number of these users participating in the system is small. The results indicate that intersection attacks remain effective even when the system has a large user base or employs longer communication rounds. This effectiveness also holds true when countermeasures such as cover traffic or delayed messages are used. While delaying messages for several hours reduces the effectiveness of intersection attacks more than using cover traffic, neither method is able to completely prevent these attacks.

3.1.4 *Mitigation of Intersection Attacks in Anonymous Microblogging*

In Chapter P4, we proposed a protocol designed to effectively protect users of anonymous microblogging systems from de-anonymization through intersection attacks. This protocol provides very strong anonymity protection through indistinguishability. However, to achieve indistinguishability, it does not require, as in related work [2, 39], that all users behave identically or remain online continuously—requirements that are impractical and lead to excessive bandwidth overhead. Instead, it groups users with similar publishing behaviors into sets, ensuring indistinguishability only within these sets rather than across the entire user base. Users within each set communicate according to a specific schedule, which can be configured to balance latency and bandwidth

1 https://x.com/ last accessed 16 October 2024
2 https://www.reddit.com/ last accessed 16 October 2024

overhead. To maintain protection above a certain level, the set size must exceed a specific threshold. Evaluation results show that the protocol effectively protects users from intersection attacks without introducing high bandwidth or latency overhead. Moreover, the protocol manages to sustain a slow decline in the size of the anonymity set over time, when there is churn in the network (i.e., when some users do not adhere to schedules).

3.1.5 *Understanding Privacy & Anonymity Perceptions of Medical Data Donation Apps*

In Chapter P5, we conducted a qualitative study to investigate the privacy and anonymity perceptions of non-users of medical data donation apps. Our study revealed that participants struggled to understand how these apps operate, highlighting the need for clearer communication. Participants preferred apps developed by research institutes over those from commercial entities, desired control over their data, and expressed concerns about technical burdens, data misuse, breaches, and discrimination. They lacked a clear understanding of data sensitivity and privacy protection methods but demanded strong privacy guarantees, particularly anonymity. Participants with less knowledge of privacy and anonymity protections were less willing to donate their data. We identified gaps between participants' expectations and needs and what existing medical data donation apps currently offer. Furthermore, we provided guidance for the development of future user-centric, privacy-preserving medical data donation apps.

This section explains how the contributions in this dissertation can be extended to other settings. It also explores the implications of these contributions, focusing on how they empower both research and users. Moreover, it discusses the potential risks of misuse, particularly in the context of anonymous social networking.

3.2.1 *Extension to Other Scenarios or Settings*

The contributions of this dissertation were designed for the specific contexts of social networking and crowdsourcing. However, they can be applied to or further expanded in scenarios and settings beyond those initially addressed:

1) Extending 2PPS and Anonify to Other Settings: 2PPS was specifically designed for online social networking applications, but it can be extended to any application requiring anonymity in communication settings that follow the publish/subscribe model. Examples of further applications could include IoT-based scenarios where devices in smart cities can exchange data anonymously on topics related to traffic and environmental monitoring.

Similarly, Anonify was developed for anonymous medical data donation. However, it can also be applied to other scenarios that involve collecting sensitive information from individuals to create public datasets. Such examples include surveys on topics like workplace satisfaction, educational experiences, social media usage, and consumer product preferences. Additionally, it is suitable for data-sharing situations that necessitate one-time data submissions, as well as those requiring periodic information sharing, as in longitudinal studies.

2) Broadening Intersection Attack Variants and Mitigation: The variants of intersection attacks proposed in Chapter P3 were initially developed to specifically target anonymous microblogging systems; however, they can be extended to other anonymous public group communication scenarios. In this context, the user-pseudonym linking attack can even be applied to any system relying on a pseudonym-based messaging pattern, where users' messages are posted under fake iden-

tifiers (pseudonyms), such as in public forums or open chat rooms. The user-topic linking attack can also be used against any system that relies on a topic-based messaging pattern, where updates or news about a certain topic can be shared without attaching pseudonyms. For example, this includes systems that allow users to anonymously report issues on topics without using pseudonyms.

Our mitigation protocol was originally proposed to address intersection attacks in the context of anonymous microblogging, but it can also be applied to other scenarios that involve a pseudonym-based messaging pattern, like the ones mentioned earlier.

3.2.2 *Empowering Research while Preserving Privacy*

Anonify guarantees strong anonymity for data donors at both communication and data levels which may increase people's willingness to donate data [8, 72, 73], resulting in collecting more comprehensive datasets. Such datasets can accelerate medical research by helping scientists identify trends and discover new treatments, which may lead to significant breakthroughs in personalized medicine and disease prevention.

Unlike many anonymous medical data donation systems that only permit one-time submissions, Anonify facilitates longitudinal studies by allowing for the collection of data from users at regular intervals while preserving donor anonymity. This capability enables researchers to link multiple data points from the same donor over time without being able to determine their true identity. This empowers research by enabling the study of changes in health behaviors and conditions over time.

As we discussed in Section 2.2.4, ensuring anonymity often comes at the cost of data utility. Despite strong privacy protections, Anonify can maintain high data utility, as demonstrated by the results of various machine learning algorithms applied to datasets anonymized by Anonify (see Chapter P2). This ensures that anonymized data remains relevant and useful for analysis, enabling researchers to draw meaningful conclusions without sacrificing the privacy of data donors.

3.2.3 *Empowering Users in Different Contexts*

In an era of constant threats to privacy and expression, empowering users is crucial. This section explores how 2PPS can assist users in both oppressive and democratic contexts.

Oppressive Regimes: A solution like 2PPS can help users in oppressive countries by providing strong anonymity, thereby protecting them from government surveillance. Further, 2PPS enables individuals to express dissent, share ideas, and connect with like-minded people without fear of retaliation. Users can organize resistance movements, discuss sensitive topics, and raise awareness about human rights abuses in a safe environment. By protecting their identities, 2PPS can allow open dialogue, help users build supportive networks, and collaborate on strategies for change, ultimately empowering them against oppressive regimes.

Democratic Countries: In democratic countries, in contrast, 2PPS can empower users by further enhancing privacy and freedom of expression. It can encourage participation in discussions that might be sensitive or even controversial. This could lead to more honest exchanges about topics, such as politics, social justice, and personal beliefs, free from the pressure of social judgment or backlash. Additionally, using a solution like 2PPS could empower individuals to explore diverse perspectives and engage with marginalized voices that might otherwise be silenced in mainstream discourse. It can also facilitate the sharing of personal stories and experiences, helping to build empathy and understanding across different communities. Moreover, it can be used by those dealing with stigmatized issues, such as mental health challenges or discrimination, allowing users to seek support and advice without fear of exposure.

3.2.4 *The Impact of Efficiency*

As shown in the evaluation in Chapter P1, 2PPS is efficient regarding both latency and bandwidth overhead. This is vital because anonymity protection should be accessible to everyone, not just those with considerable resources or the ability to tolerate long delays. Minimizing bandwidth overhead can be particularly essential in bandwidth-constrained

environments or areas with limited internet infrastructure. Additionally, reduced bandwidth usage can lower costs for users, as it leads to lower data consumption—an important factor in many regions worldwide, such as rural areas in India and regions in sub-Saharan Africa. Low latency in anonymous social networking can also be crucial, especially for users who need instant communication, such as the immediate publishing of critical content.

3.2.5 *Potential Misuse of Anonymous Social Networking*

Although this dissertation demonstrates the importance of anonymity in public group communication, it is essential to differentiate between contexts where anonymity is beneficial and those where it can lead to misuse, particularly in the realm of social networking.

In situations such as whistleblowing, mental health discussions, or activism, anonymity serves a vital purpose. It empowers individuals to express their views or seek support without fear and promotes the sharing of diverse perspectives.

However, anonymity can also facilitate harmful behaviors, such as cyberbullying, harassment, and the dissemination of misinformation [62]. In these contexts, users may exploit anonymity to evade accountability for their illegal or immoral actions, leading to "toxic" online environments where malicious comments and behaviors thrive unchecked. Furthermore, anonymity can enable criminal activities like scams and identity theft, as individuals hide behind fake profiles.

Thus, while anonymity offers significant benefits in certain situations, it also raises critical concerns about safety and responsibility in online interactions. Addressing the issue of misuse in anonymous social networking is important yet challenging, as detecting misconduct and criminal activity without compromising user anonymity can be hard.

3.3 FUTURE WORK

We see the following avenues for future improvement in the domain of anonymous public group communication:

3.3.1 *Focusing Even More on the Human Factors Aspect*

In the anonymity domain, including anonymous public group communication, the human factors aspect is often overlooked by researchers who tend to focus primarily on the technical robustness and security of the systems. While achieving strong anonymity guarantees is essential, neglecting human factors can lead to systems that are too complex or unintuitive for the average user. This gap can result in poor adoption rates, user errors, or misuse that undermines the system's intended privacy protections. Moreover, anonymity systems "love company" — they rely on larger user bases to create bigger anonymity sets, where individuals are better hidden among many other users. If the usability of the system discourages people from using it, the anonymity set shrinks, and the overall effectiveness of the system is compromised. Therefore, considering human factors is essential for attracting more users and enhancing anonymity for every user. We advocate for more research into user perceptions and behavior in the context of anonymous public group communication systems. Understanding user preferences, needs, expectations, and misconceptions can lead to systems that balance strong security with usability, promoting broader adoption and effective use.

3.3.2 *Addressing the Need for Real-World Data*

Designing effective digital systems requires making various assumptions about the systems' operation, goals, users, and the data they handle. The success of these systems relies heavily on the accuracy of the assumptions made about them and the quality of the data used in their development and evaluation. In the area of anonymous public group communication, there is a severe shortage of high-quality, real-world data. For instance, while we have been able to gather data on publishing behaviors from Twitter and Reddit, we lack data on user

subscription behaviors and have struggled to find real-world datasets for other public group communication scenarios. This lack of realistic data often results in flawed or unrealistic assumptions when designing anonymity systems. To address this issue and enhance both current and future anonymity systems, there is an urgent need to collect large, high-quality, and detailed datasets that accurately reflect various aspects of user behavior. For ethical considerations, these datasets should be carefully anonymized to ensure that the information cannot be traced back to real individuals.

3.3.3 *Enabling Customization*

Users of anonymous public group communication systems come from diverse backgrounds and face varying levels of risk. For instance, a journalist or political dissident in an oppressive regime may need the highest level of anonymity to avoid detection and persecution, as their adversary could be a powerful state actor with extensive surveillance capabilities. In contrast, a casual user from a democratic country with freedom of speech might only seek anonymity from other users in the system or from the system itself. Since higher levels of anonymity often result in performance trade-offs, such as increased latency and bandwidth overhead, it may be unnecessary for casual users to opt for the strongest protections when their risks are lower. Therefore, there is a need for customizability in anonymous public group communication systems. Allowing individuals to adjust their level of anonymity and performance provides a balance between security, usability, and efficiency. This flexibility ensures that users—from journalists and activists to casual users—can tailor the system to meet their specific needs and risks, adapting to their unique situations while maintaining appropriate protection. Research we conducted in [29] can serve as a foundation for efforts to design effective customizable anonymity solutions for public group communication scenarios.

REFERENCES

[1] Timothy G Abbott, Katherine J Lai, Michael R Lieberman, and Eric C Price. “Browser-based attacks on Tor.” In: *International*

Workshop on Privacy Enhancing Technologies. Springer. 2007, pp. 184–199.

[2] Ittai Abraham, Benny Pinkas, and Avishay Yanai. "Blinder–Scalable, Robust Anonymous Committed Broadcast." In: *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 2020, pp. 1233–1252.

[3] Bechir Alaya, Lamri Laouamer, and Nihel Msilini. "Homomorphic encryption systems statement: Trends and challenges." In: *Computer Science Review* 36 (2020), p. 100235.

[4] Sebastian Angel and Srinath Setty. "Unobservable communication over fully untrusted infrastructure." In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. 2016, pp. 551–569.

[5] Roberto J Bayardo and Rakesh Agrawal. "Data privacy through optimal k-anonymization." In: *21st International conference on data engineering (ICDE'05)*. IEEE. 2005, pp. 217–228.

[6] O. Berthold et al. "Dummy traffic against long term intersection attacks." In: *International Workshop on Privacy Enhancing Technologies*. Springer. 2002, pp. 110–128.

[7] Oliver Berthold and Heinrich Langos. "Dummy traffic against long term intersection attacks." In: *Privacy Enhancing Technologies: Second International Workshop, PET 2002 San Francisco, CA, USA, April 14–15, 2002 Revised Papers 2*. Springer. 2003, pp. 110–128.

[8] Richard Brown, Elizabeth Sillence, Lynne Coventry, Emma Simpson, Jo Gibbs, Shema Tariq, Abigail C. Durrant, and Karen Lloyd. "Understanding the attitudes and experiences of people living with potentially stigmatised long-term health conditions with respect to collecting and sharing health and lifestyle data." In: *Digital health* 8 (2022), p. 20552076221089798.

[9] Carole Cadwalladr and Emma Graham-Harrison. *Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach*. https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election (Accessed 06-July-2024). 2018.

[10] Tânia Carvalho, Nuno Moniz, Pedro Faria, and Luís Antunes. "Survey on privacy-preserving techniques for data publishing." In: *arXiv preprint arXiv:2201.08120* (2022).

[11] David Chaum. "The dining cryptographers problem: Unconditional sender and recipient untraceability." In: *Journal of cryptology* 1 (1988), pp. 65–75.

[12] David L Chaum. "Untraceable electronic mail, return addresses, and digital pseudonyms." In: *Communications of the ACM* 24.2 (1981), pp. 84–90.

[13] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. "Private information retrieval." In: *Journal of the ACM (JACM)* 45.6 (1998), pp. 965–981.

[14] M Cilia, M Antollini, C Bornhövd, and A Buchmann. "Dealing with heterogeneous data in pub/sub systems: The Concept-Based approach." In: *Proc. 3rd Int'l Workshop on Distributed Event-Based Systems*. 2004, pp. 26–31.

[15] Henry Corrigan-Gibbs, Dan Boneh, and David Mazières. "Riposte: An anonymous messaging system handling millions of users." In: *2015 IEEE Symposium on Security and Privacy*. IEEE. 2015, pp. 321–338.

[16] Henry Corrigan-Gibbs and Bryan Ford. "Dissent: accountable anonymous group messaging." In: *Proceedings of the 17th ACM conference on Computer and communications security*. 2010, pp. 340–350.

[17] George Danezis. "Statistical disclosure attacks: Traffic confirmation in open environments." In: *Security and Privacy in the Age of Uncertainty: IFIP TC11 18 th International Conference on Information Security (SEC2003) May 26–28, 2003, Athens, Greece 18*. Springer. 2003, pp. 421–426.

[18] George Danezis and Andrei Serjantov. "Statistical Disclosure or Intersection Attacks on Anonymity Systems." In: *Information Hiding, 6th International Workshop, IH 2004, Toronto, Canada, May 23-25, 2004, Revised Selected Papers*. Vol. 3200. Lecture Notes in Computer Science. Springer, 2004, pp. 293–308.

[19] Debajyoti Das, Sebastian Meiser, Esfandiar Mohammadi, and Aniket Kate. "Anonymity trilemma: Strong anonymity, low bandwidth overhead, low latency-choose two." In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2018, pp. 108–126.

[20] Jörg Daubert, Mathias Fischer, Tim Grube, Stefan Schiffner, Panayotis Kikiras, and Max Mühlhäuser. "AnonPubSub: Anonymous publish-subscribe overlays." In: *Computer Communications* 76 (2016), pp. 42–53.

[21] Jörg Daubert, Tim Grube, Max Mühlhäuser, and Mathias Fischer. "Internal attacks in anonymous publish-subscribe P2P overlays." In: *2015 International Conference and Workshops on Networked Systems (NetSys)*. IEEE. 2015, pp. 1–8.

[22] Roger Dingledine, Nick Mathewson, Paul F Syverson, et al. "Tor: The second-generation onion router." In: *USENIX security symposium*. Vol. 4. 2004, pp. 303–320.

[23] Cynthia Dwork. "Differential privacy." In: *International colloquium on automata, languages, and programming*. Springer. 2006, pp. 1–12.

[24] Cynthia Dwork. "Differential privacy: A survey of results." In: *International conference on theory and applications of models of computation*. Springer. 2008, pp. 1–19.

[25] Matthew Edman and Bülent Yener. "On anonymity in an electronic society: A survey of anonymous communication systems." In: *ACM Computing Surveys (CSUR)* 42.1 (2009), pp. 1–35.

[26] Benjamin CM Fung, Ke Wang, Rui Chen, and Philip S Yu. "Privacy-preserving data publishing: A survey of recent developments." In: *ACM Computing Surveys (Csur)* 42.4 (2010), pp. 1–53.

[27] Sarah Gaballah, Thanh Hoang Long Nguyen, Lamya Abdullah, Ephraim Zimmer, and Max Mühlhäuser. "Mitigating Intersection Attacks in Anonymous Microblogging." In: *Proceedings of the 18th International Conference on Availability, Reliability and Security*. 2023.

[28] Sarah Abdelwahab Gaballah, Lamya Abdullah, Mina Alishahi, Thanh Hoang Long Nguyen, Ephraim Zimmer, Max Mühlhäuser, and Karola Marky. "Anonify: Decentralized Dual-level Anonymity for Medical Data Donation." In: *Proceedings on Privacy Enhancing Technologies* 3 (2024), pp. 94–108.

[29] Sarah Abdelwahab Gaballah, Lamya Abdullah, Max Mühlhäuser, and Karola Marky. "Let the Users Choose: Low Latency or Strong Anonymity? Investigating Mix Nodes with Paired Mixing Techniques." In: *Proceedings of the 19th International Conference on Availability, Reliability and Security*. 2024.

[30] Sarah Abdelwahab Gaballah, Lamya Abdullah, Minh Tung Tran, Ephraim Zimmer, and Max Mühlhäuser. "On the effectiveness of intersection attacks in anonymous microblogging." In: *Nordic Conference on Secure IT Systems*. Springer. 2022, pp. 3–19.

[31] Sarah Abdelwahab Gaballah, Lamya Abdullah, Ephraim Zimmer, Sascha Fahl, Max Mühlhäuser, and Karola Marky. ""It's Not My Data Anymore": Exploring Non-Users' Privacy Perceptions of Medical Data Donation Apps." In: *Proceedings on Privacy Enhancing Technologies* 1 (2025), pp. 654–670.

[32] Sarah Abdelwahab Gaballah, Christoph Coijanovic, Thorsten Strufe, and Max Mühlhäuser. "2PPS—publish/subscribe with provable privacy." In: *2021 40th international symposium on reliable distributed systems (SRDS)*. IEEE. 2021, pp. 198–209.

[33] George Giakkoupis, Rachid Guerraoui, Arnaud Jégou, Anne-Marie Kermarrec, and Nupur Mittal. "Privacy-conscious information diffusion in social networks." In: *International Symposium on Distributed Computing*. Springer. 2015, pp. 480–496.

[34] Ian Goldberg. "Improving the robustness of private information retrieval." In: *2007 IEEE Symposium on Security and Privacy (SP'07)*. IEEE. 2007, pp. 131–148.

[35] Jamie Hayes, Carmela Troncoso, and George Danezis. "TASP: Towards anonymity sets that persist." In: *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society*. 2016, pp. 177–180.

[36] Rob Jansen, Florian Tschorsch, Aaron Johnson, and Björn Scheuermann. *The sniper attack: Anonymously deanonymizing and disabling the Tor network*. Tech. rep. Office of Naval Research Arlington VA, 2014.

[37] Aaron Johnson, Chris Wacek, Rob Jansen, Micah Sherr, and Paul Syverson. "Users get routed: Traffic correlation on Tor by realistic adversaries." In: *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. 2013, pp. 337–348.

[38] Xiangjie Kong, Xiaoteng Liu, Behrouz Jedari, Menglin Li, Liangtian Wan, and Feng Xia. "Mobile crowdsourcing in smart cities: Technologies, applications, and future challenges." In: *IEEE Internet of Things Journal* 6.5 (2019), pp. 8095–8113.

[39] A. Kwon et al. "Riffle: An efficient communication system with strong anonymity." In: *Proceedings on Privacy Enhancing Technologies* (2016).

[40] Albert Kwon, Henry Corrigan-Gibbs, Srinivas Devadas, and Bryan Ford. "Atom: Horizontally scaling strong anonymity." In: *Proceedings of the 26th Symposium on Operating Systems Principles*. 2017, pp. 406–422.

[41] Hengky Latan, Charbel Jose Chiappetta Jabbour, and Ana Beatriz Lopes de Sousa Jabbour. "Social media as a form of virtual whistleblowing: Empirical evidence for elements of the diamond model." In: *Journal of Business Ethics* 174 (2021), pp. 529–548.

[42] Jiexing Li, Yufei Tao, and Xiaokui Xiao. "Preservation of proximity privacy in publishing numerical sensitive data." In: *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. 2008, pp. 473–486.

[43] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. "t-closeness: Privacy beyond k-anonymity and l-diversity." In: *2007 IEEE 23rd international conference on data engineering*. IEEE. 2006, pp. 106–115.

[44] Dong Lin, Micah Sherr, and Boon Thau Loo. "Scalable and anonymous group communication with MTor." In: *Proceedings on Privacy Enhancing Technologies* (2016).

[45] Ewen MacAskill and Gabriel Dance. *NSA files decoded: Edward Snowden's surveillance revelations explained*. https://www.theguardian.com/world/interactive/2013/nov/01/snowden-nsa-files-surveillance-revelations-decoded (Accessed 06-July-2024). 2013.

[46] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. "l-diversity: Privacy beyond k-anonymity." In: *Acm transactions on knowledge discovery from data (tkdd)* 1.1 (2007), 3–es.

[47] National Coalition Against Censorship. *EFF Debunks NSA Mass Surveillance Apologists*. https://ncac.org/news/blog/eff-debunks-nsa-mass-surveillance-apologists (Accessed 06-July-2024).

[48] Mehran Alidoost Nia and Antonio Ruiz-Martinez. "Systematic literature review on the state of the art and future research work in anonymous communications systems." In: *Computers & electrical engineering* 69 (2018), pp. 497–520.

[49] Iyiola E Olatunji, Jens Rauch, Matthias Katzensteiner, and Megha Khosla. "A review of anonymization for healthcare data." In: *Big data* (2022).

[50] Babajide Osatuyi. “Information sharing on social media sites.” In: *Computers in human behavior* 29.6 (2013), pp. 2622–2631.

[51] European Parliament. *Regulation (EU) 2016/679*. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:32016R0679 (Accessed 02-July-2024). 2016.

[52] Van L Parsons. “Stratified sampling.” In: *Wiley StatsRef: Statistics Reference Online* (2014), pp. 1–11.

[53] Thomas Paul, Antonino Famulari, and Thorsten Strufe. “A survey on decentralized online social networks.” In: *Computer Networks* 75 (2014), pp. 437–452.

[54] Sai Teja Peddinti, Keith W Ross, and Justin Cappos. “" On the internet, nobody knows you’re a dog" a twitter case study of anonymity in social networks.” In: *Proceedings of the second ACM conference on Online social networks*. 2014, pp. 83–94.

[55] Andreas Pfitzmann and Marit Hansen. *A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management*. 2010.

[56] Andreas Pfitzmann and Marit Köhntopp. “Anonymity, unobservability, and pseudonymity—a proposal for terminology.” In: *Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability Berkeley, CA, USA, July 25–26, 2000 Proceedings*. Springer. 2001, pp. 1–9.

[57] Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. “The Text Anonymization Benchmark (TAB): A Dedicated Corpus and Evaluation Framework for Text Anonymization.” In: *Computational Linguistics* 48.4 (Dec. 2022), pp. 1053–1101. ISSN: 0891-2017.

[58] Ania M Piotrowska. “Studying the anonymity trilemma with a discrete-event mix network simulator.” In: *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*. 2021, pp. 39–44.

[59] Michael G Reed, Paul F Syverson, and David M Goldschlag. “Anonymous connections and onion routing.” In: *IEEE Journal on Selected areas in Communications* 16.4 (1998), pp. 482–494.

[60] Michael K Reiter and Aviel D Rubin. “Crowds: Anonymity for web transactions.” In: *ACM transactions on information and system security (TISSEC)* 1.1 (1998), pp. 66–92.

[61] Jian Ren and Jie Wu. "Survey on anonymous communications in computer networks." In: *Computer Communications* 33.4 (2010), pp. 420–431.

[62] Saed Rezayi, Vimala Balakrishnan, Samira Arabnia, and Hamid R Arabnia. "Fake news and cyberbullying in the modern era." In: *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE. 2018, pp. 7–12.

[63] RKI. *Corona Data Donation Project*. https://corona-datenspende.github.io/en/ (Accessed 15-July-2024). 2019.

[64] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. "A metric for distributions with applications to image databases." In: *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*. IEEE. 1998, pp. 59–66.

[65] Bharath K Samanthula, Gerry Howser, Yousef Elmehdwi, and Sanjay Madria. "An efficient and secure data sharing framework using homomorphic encryption in the cloud." In: *Proceedings of the 1st International Workshop on Cloud Intelligence*. 2012, pp. 1–8.

[66] Hossein Shafagh, Anwar Hithnawi, Lukas Burkhalter, Pascal Fischli, and Simon Duquennoy. "Secure sharing of partially homomorphic encrypted iot data." In: *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 2017, pp. 1–14.

[67] Haoyi Shi, Chao Jiang, Wenrui Dai, Xiaoqian Jiang, Yuzhe Tang, Lucila Ohno-Machado, and Shuang Wang. "Secure multi-pArty computation grid LOgistic REgression (SMAC-GLORE)." In: *BMC medical informatics and decision making* 16 (2016), pp. 175–187.

[68] Haris Smajlović, Ariya Shajii, Bonnie Berger, Hyunghoon Cho, and Ibrahim Numanagić. "Sequre: a high-performance framework for secure multiparty computation enables biomedical data sharing." In: *Genome Biology* 24.1 (2023), p. 5.

[69] Yixin Sun, Anne Edmundson, Laurent Vanbever, Oscar Li, Jennifer Rexford, Mung Chiang, and Prateek Mittal. "{RAPTOR}: Routing attacks on privacy in tor." In: *24th USENIX Security Symposium (USENIX Security 15)*. 2015, pp. 271–286.

[70] Latanya Sweeney. "k-anonymity: A model for protecting privacy." In: *International journal of uncertainty, fuzziness and knowledge-based systems* 10.05 (2002), pp. 557–570.

[71] Catherine Thorbecke. *Facebook says government requests for user data have reached all-time high*. https://abcnews.go.com/Business/facebook-government-requests-user-data-reached-time-high/story?id=66981424 (Accessed 26-July-2024). 2019.

[72] Christine Utz, Steffen Becker, Theodor Schnitzler, Florian M Farke, Franziska Herbert, Leonie Schaewitz, Martin Degeling, and Markus Dürmuth. "Apps against the spread: Privacy implications and user acceptance of COVID-19-related smartphone apps on three continents." In: *Proceedings of the 2021 chi conference on human factors in computing systems*. 2021, pp. 1–22.

[73] André Calero Valdez and Martina Ziefle. "The users' perspective on the privacy-utility trade-offs in health recommender systems." In: *International Journal of Human-Computer Studies* 121 (2019), pp. 108–121.

[74] Alan F Westin. "Privacy and freedom." In: *Washington and Lee Law Review* 25.1 (1968), p. 166.

[75] Felix Nikolaus Wirth, Tobias Kussel, Armin Müller, Kay Hamacher, and Fabian Prasser. "EasySMPC: a simple but powerful no-code tool for practical secure multiparty computation." In: *BMC bioinformatics* 23.1 (2022), p. 531.

[76] Felix Nikolaus Wirth, Thierry Meurers, Marco Johns, and Fabian Prasser. "Privacy-preserving data sharing infrastructures for medical research: systematization and comparison." In: *BMC Medical Informatics and Decision Making* 21 (2021), pp. 1–13.

[77] D. Wolinsky et al. "Hang with your buddies to resist intersection attacks." In: *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. 2013, pp. 1153–1166.

[78] David Isaac Wolinsky, Henry Corrigan-Gibbs, Bryan Ford, and Aaron Johnson. "Scalable anonymous group communication in the anytrust model." In: *European Workshop on System Security (EuroSec)*. Vol. 4. 2012.

[79] Alexander Wood, Kayvan Najarian, and Delaram Kahrobaei. "Homomorphic encryption for machine learning in medicine and bioinformatics." In: *ACM Computing Surveys (CSUR)* 53.4 (2020), pp. 1–35.

[80] Jian Xu, Wei Wang, Jian Pei, Xiaoyuan Wang, Baile Shi, and Ada Wai-Chee Fu. "Utility-based anonymization for privacy preservation with less information loss." In: *Acm Sigkdd Explorations Newsletter* 8.2 (2006), pp. 21–30.

[81] Chuan Zhao, Shengnan Zhao, Minghao Zhao, Zhenxiang Chen, Chong-Zhi Gao, Hongwei Li, and Yu-an Tan. "Secure multi-party computation: theory, practice and applications." In: *Information Sciences* 476 (2019), pp. 357–372.

Part II

PUBLICATIONS

P1

2PPS – PUBLISH/SUBSCRIBE WITH PROVABLE PRIVACY

This chapter was first published as

Sarah Abdelwahab Gaballah, Christoph Coijanovic, Thorsten Strufe, and Max Mühlhäuser. "2PPS—Publish/Subscribe with Provable Privacy." In 40th International Symposium on Reliable Distributed Systems (SRDS), pp. 198-209. © 2021 IEEE.

and is reproduced with permission from IEEE. The version of record of this article, first published in the proceedings of the 2021 40th International Symposium on Reliable Distributed Systems (SRDS), is available online at the publisher's website: https://doi.org/10.1109/SRDS53918.2021.00028

Contribution Statement: I led the idea generation, conceptualization, and development of the 2PPS protocol, as well as implementing the protocol, conducting experiments, and performing data analysis. Christoph Coijanovic contributed to the paper by providing formalization and security analysis of 2PPS privacy goals and discussing the protocol's average cover ratio in the network overhead subsection of the performance evaluation section. Additionally, Christoph and I collaborated to enhance the resilience of the 2PPS protocol against active attacks. Further, both contributed to the writing of the publication. Thorsten Strufe and Max Mühlhäuser provided valuable discussions and insights on key aspects of the work, including feedback on earlier versions of the paper.

2021 40th International Symposium on Reliable Distributed Systems (SRDS)

2PPS – Publish/Subscribe with Provable Privacy

Sarah Abdelwahab Gaballah
TU Darmstadt
gaballah@tk.tu-darmstadt.de

Christoph Coijanovic
Karlsruhe Institute of Technology
christoph.coijanovic@kit.edu

Thorsten Strufe
Karlsruhe Institute of Technology
thorsten.strufe@kit.edu

Max Mühlhäuser
TU Darmstadt
max@tk.tu-darmstadt.de

***Abstract*—Publish/Subscribe systems like Twitter and Reddit let users communicate with many recipients without requiring prior personal connections. The content that participants of these systems publish and subscribe to is typically public, but they may nevertheless wish to remain anonymous. While many existing systems allow users to omit explicit identifiers, they do not address the obvious privacy risks of being associated with content that may contain a wide range of sensitive information.**

We present 2PPS (*Twice-Private Publish-Subscribe*), the first pub/sub protocol to deliver strong provable privacy protection for both publishers and subscribers, leveraging Distributed Point Function-based secret sharing for publishing and Private Information Retrieval for subscribing. 2PPS does not require trust in other clients and its privacy guarantees hold as long as even a single honest server participant remains. Furthermore, it is scalable and delivers latency suitable for microblogging applications.

A prototype implementation of 2PPS can handle 100,000 concurrent active clients with 5 seconds end-to-end latency and significantly lower bandwidth requirements than comparable systems.

***Index Terms*—privacy, anonymity, publish/subscribe, private information retrieval**

I. Introduction

Consider the immense popularity of services like Twitter, Reddit, and Telegram. All can be classified as implementing the Publish/Subscribe (pub/sub) messaging pattern: Messages are *published* to certain *topics* (e.g., hashtags for Twitter or channels for Telegram). Users can freely *subscribe* to topics they are interested in and will receive corresponding messages. The service acts as an intermediary broker between publisher and subscribers and is responsible for managing subscriptions, sorting received messages by topic, and forwarding them to the intended subscribers.

One particularly interesting use of pub/sub systems is the organization of volunteering, political involvement, and activism. Pub/sub lends itself to this setting since it allows large numbers of people to connect without having a prior personal relationship. Protesters in Iraq, Hong Kong, and Belarus have been using Telegram and FireChat for this purpose [1]–[3]. However, the use of conventional systems can leak valuable *metadata* to an adversary:

- An activist who is found out to be publishing or subscribing to a regime-critical topic may be facing serious legal consequences.
- If the regime finds out how many users are subscribed to a critical topic, it can determine the size of the activists' movement and deploy an overwhelming police force at the next protest.

Telegram and FireChat do not protect this kind of metadata [4]. This is where our proposed protocol, 2PPS, comes in: It offers strong provable privacy protection for both publishers and subscribers in an open pub/sub setting. History shows that a state-level adversary can have access to immense resources [5]. Thus, we aim to protect against an adversary who may not only corrupt users and servers but also observe and interfere with traffic globally. While the example of political activists impressively motivates the need for private pub/sub communication, it is not the only possible use for 2PPS. Service providers can infer sensitive information such as health problems, financial status, or sexual preferences by observing which topics a user is active in.

At a high level, 2PPS reaches its goal as follows: The broker's functionality is distributed over multiple servers, where privacy is ensured as long as at least one arbitrary server does not collude with the adversary. While the adversary inherently learns which messages are published, since she can join arbitrary groups herself, she cannot learn any further information (e.g., who publishes which message). This is achieved by using *Distributed Point Function* (DPF)-based secret sharing. Compared to prior work [6], [7], we present an improved secret sharing approach that also protects against active interference. Subscribers protect their privacy by using *Private Information Retrieval* (PIR) for receiving messages.

To the best of our knowledge, no existing protocol can provide open pub/sub communication with strong provable privacy guarantees for both publishers and subscribers. Some protocols require trusted group members [8]–[10] or trusted execution environments [11]. Others don't provide both sender and receiver anonymity [10] or don't target worst-case protection [12]. Additionally, some of them are vulnerable to traffic analysis attacks [13]. PIR-based protocols offer strong cryptographic security guarantees and hide metadata efficiently [6], [7], [14]. However, the majority of these protocols support either point-to-point communication [7] or broadcasting [6],

This work was supported by funding of the German Research Foundation (DFG), research grant 317688284 and by funding of the Helmholtz Association (HGF) through the Competence Center for Applied Security Technology (KASTEL)

2575-8462/21/$31.00 ©2021 IEEE
DOI 10.1109/SRDS53918.2021.00028

[14]. PIR-based protocols that support selective multicast communication either provide strong receiver anonymity but weak sender anonymity [9], or do not scale well [15].
Designing an anonymous communication protocol always requires a trade-off between privacy protection, trust, and overhead [16]. However, we show that 2PPS, despite strong privacy protection and minimal trust requirements, manages to keep the overhead for clients at a reasonable level:

- It incurs a latency of 25 s to handle one million users where each client submits a 160 B message and receives a 10 KB block of messages.
- For 50 subscriptions per client, 2PPS requires 300× less bandwidth compared to broadcasting systems such as Riposte [6] and Blinder [14].

Contributions: In this paper, we make the following contributions:

- We introduce 2PPS, a new anonymous publish/subscribe protocol by combining private writing using Distributed Point Functions (DPF) and private reading using Information-Theoretic Private Information Retrieval (IT-PIR).
- We give formal proof to show that the 2PPS protocol reaches our stated goals of Publisher- and Subscriber Unobservability.
- We provide an evaluation of 2PPS that demonstrates its efficiency in terms of latency and bandwidth.

II. Model and Goals

A. Protocol Overview

2PPS implements the publish/subscribe model: When *publishing* a message, the sending user (i.e., the publisher) specifies a *topic* the message belongs to. The protocol then delivers this message to all *subscribers* of this topic. We assume an open environment, meaning users can freely subscribe to and unsubscribe from any topic they wish.
The 2PPS network consists of n clients and N servers with $n \gg N$. Each server stores a full copy of two databases, a *write* database D_w and a *read* database D_r . All servers collectively maintain the contents of these databases. The read database D_r is partitioned into a set of ℓ_r equal-sized blocks, with a publicly known topic for each block. The write database D_w is partitioned into ℓ_w equal-sized blocks, each sized to hold one message.
Similar to related protocols [6], [15], [17], communication in 2PPS occurs in rounds to defend against traffic analysis attacks. Each round is split into three distinct phases (Figure 1 depicts these phases at a high level). During the first phase, each client deposits exactly one message into the write database D_w, which is shared among the N servers using secret sharing. If a client has no real message to send, it generates a cover message. After the first phase has concluded, the servers collaborate to reveal all messages simultaneously. During the second phase, cover messages are discarded and the remaining messages are sorted into their corresponding topic-block of the read database. Finally, in the third phase, clients anonymously retrieve the messages from their subscribed topic.

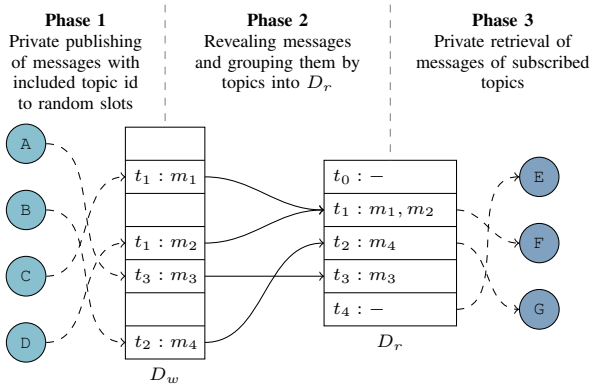


Fig. 1: Server databases (D_w and D_r) and general protocol flow.

B. Threat Model

2PPS assumes a strong adversary $\mathcal{A}$, whose goal is to compromise the privacy of honest users. $\mathcal{A}$ is assumed to be in control of all network links. Thus, she may not only passively observe all traffic on every link, but also insert, delay, drop, time, and modify arbitrary packets. Further, $\mathcal{A}$ may corrupt $N-1$ servers and an arbitrary number of clients. Since we assume open groups, $\mathcal{A}$ may join all groups as a subscriber through corrupted clients. We assume that honest clients and servers behave as specified by the 2PPS protocol.

C. Security Goals

2PPS aims to achieve the formal privacy notions of *Publisher Unobservability* and *Subscriber Unobservability*. Kuhn et al. present a set of game-based privacy notions for unicast communication [18], which we adapt for the pub/sub scenario.
Each notion is defined by a game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$:

1) $\mathcal{C}$ chooses a random challenge bit $b \in \{0,1\}$.
2) $\mathcal{A}$ submits a challenge consisting of two self-chosen scenarios (S_0, S_1) to $\mathcal{C}$. Each scenario contains a number of communications (p, m, t), where p denotes the publisher, m the message, and t the topic that p sends m to.
3) $\mathcal{C}$ checks the received challenge for validity and, if valid, simulates the protocol execution of the communications contained in S_b.
4) Based on his abilities, $\mathcal{A}$ gets to observe and interact with the protocol execution.
5) $\mathcal{A}$ determines which of his scenarios was chosen and submits his guess $b' \in \{0,1\}$ to $\mathcal{C}$. $\mathcal{A}$ wins if $b = b'$.

Steps 2-5 can be repeated. Instead of a challenge, the adversary can also submit a *subscription update*. The subscription update specifies for each client the topics she is subscribed to in each scenario. If differences in the communications between the two batches lead to differences in protocol behavior that $\mathcal{A}$ can

observe, then $\mathcal{A}$ gains an advantage over randomly guessing the chosen batch.

A concrete privacy notion defines which information is allowed to leak to the adversary and which should be protected by the protocol. Information that is allowed to leak may not differ between batches to ensure that $\mathcal{A}$ does not gain an unfair advantage when trying to distinguish. If $\mathcal{A}$ can still determine which of his batches was submitted to the protocol with a non-negligible advantage over random guessing, the protocol has failed to protect the information it was supposed to protect and therefore does not reach this privacy notion.

Publisher Unobservability: With publisher unobservability, the protocol aims to hide any information about active publishers. Implicitly, information about messages, topics, and subscribers is not protected. Thus, $\mathcal{A}$ is required to submit batches that only differ in the publisher of each communication: Le the ith communication of the submitted first batch be $(\mathbf{p_0^i}, t_0^i, m_0^i)$ where topic t_0^i has subscribers $r_0^{i,0}, \ldots, r_0^{i,n}$. Then, the ith communication of the submitted second batch has be of form $(\mathbf{p_1^i}, t_0^i, m_0^i)$ where t_0^i also has to have subscribers $r_0^{i,0}, \ldots, r_0^{i,n}$. If $\mathcal{A}$ can determine which of his batches was executed with a non-negligible advantage despite this restriction, the protocol does not reach publisher unobservability.

Subscriber Unobservability: Analogous to the publisher variant, subscriber unobservability aims to hide any information about active subscribers. Since information about publishers, messages, and topics is allowed to leak, $\mathcal{A}$ has to submit the *same* communications in both batches. However, which clients are subscribed to which topics may vary between the two batches of a challenge.

III. 2PPS Architecture

This section describes the 2PPS protocol in greater detail. Section III-A presents the anonymous publishing phase, Section III-B the management of published requests and Section III-C the anonymous subscription phase.

A. Phase 1: Anonymous Publishing.

Assume that client Alice wants to publish some message m to topic t without the adversary being able to link the message to her. We start by introducing a simple but inefficient method to hide sender identities from malicious servers. Then we improve the efficiency of this method and finally extend it to also protect against adversaries that are in control of the network.

Naïvely, *secret sharing* [19] can be employed for anonymous publishing: First, Alice computes a vector w of the same length as the write database D_w, which contains $(t \mid m)$ at a random index and 0 everywhere else. She then computes N secret shares $w_1, \ldots, w_N$ with the following properties:

1) $\sum_{i=1}^{N} w_i = w$
2) Any combination of $N-1$ secret shares does not reveal any information about $(t \mid m)$ or the index at which $(t \mid m)$ is located.

Alice distributes the shares to the servers, where the ith server S_i receives w_i. S_i then adds w_i to its read database state D_w^i:

$$D_w^i \leftarrow D_w^i + w_i$$

To hide sending frequencies, all clients are required to publish *exactly* one message per round. If the client has no "real" message to send, she may send a message consisting only of zeroes to a random topic as cover. After processing requests from multiple clients, the servers can collaborate to compute a combined database $D_w = \sum_i D_w^i$. As long as every client chose a unique index for her messages, D contains all original messages.

This approach is quite inefficient: For every write request, a vector with the same size as the database has to be sent. To address this issue, Riposte [6] suggested the use of distributed point functions (DPF).

Definition 1 (DPF): Let $f_{i^*,m} : \{0, \ldots, \ell_w\} \mapsto \mathbb{F}$ be a point function with

$$f_{i^*,m}(i) = \begin{cases} m & \text{for } i = i^* \\ 0 & \text{for } i \in \{0, \ldots, \ell\} \setminus i^* \end{cases}$$

$f_A, f_B : \{0, \ldots, \ell_w\} \mapsto \mathbb{F}$ are distributed point functions of $f_{i^*,m}$, iff

1) Neither f_A nor f_B by themselves reveal anything about m or i^*
2) $\forall\, i \in \{0, \ldots, \ell\} : f_A(i) + f_B(i) = f_{i^*,m}(i)$

DPFs can be used to *compress* the shares sent to the servers from the naïve approach: Alice runs $\mathsf{GenDPF}(t \mid m)$, which generates N DPF-shares $f_1, \ldots, f_N$ that contain $(t \mid m)$ at a random index. These shares are distributed among the servers, with server S_i receiving f_i. Server S_i can derive w_i by evaluating $f_i(j)$ at every point $j \in \{0, \ldots, \ell_w\}$. Current research states that sending a DPF share instead of w_i directly reduces the communication cost to $O(\lambda \cdot \log \ell_w + \log |(t \mid m)|)$ bits where λ is the security parameter [20].

As is, this approach protects against malicious servers, but not against stronger adversaries, who are also in control of the network: $\mathcal{A}$ could simply intercept Alice's shares before they reach the servers and combine them to reveal Alice's topic and message. To protect, Alice can encrypt each share with the receiving server's public key before sending it. Since we assume at least one honest server, $\mathcal{A}$ cannot gain access to all shares. Due to the public nature of messages in 2PPS, there are further active attacks possible:

Replay: To link Alice to her message by replay, $\mathcal{A}$ saves all shares Alice sends in a given round and all messages that are revealed in the same round. In the next round, $\mathcal{A}$ inserts the saved shares into the traffic. the servers will add them to their D_w state. $\mathcal{A}$ can identify which messages Alice has sent by observing which identical message was revealed in both rounds. To detect replayed messages, Alice includes a current timestamp with every share (inside of the encryption layer). An honest server can check the time stamp for currentness and refuse further participation in this round if this check fails. The share-encryption also prevents $\mathcal{A}$ from selectively

modifying the timestamp. $\mathcal{A}$ could also replay the shares in the same round as the original shares, which would corrupt Alice's message. However, this attack is easily detectable by the honest server, since it has access to all shares of the current round at once and can check for duplicates.

Modification: Our proposed protection against replay attacks also enables honest servers to detect if a received request was modified by $\mathcal{A}$. We assume that encryption used to protect the share and the timestamp provides *diffusion*, i.e., ensures that any change of a ciphertext leads to widespread and unforeseeable changes of the plaintext. Thus, any modification of a request leads to a significant change of the included timestamp with overwhelming probability. The honest server detects this invalid timestamp and refuses further participation in the current round.

Drop & Delay: Attacks based on dropping and delaying messages are both very common and hard to avoid in anonymous communication [21]. If a powerful adversary can drop the requests of all but one client, then he can unambiguously link this client to her messages once the shares are combined. A less powerful adversary might have insider knowledge of a message that will be sent in a given round. If he drops the request of the suspected sender and the message is not published, his suspicion is confirmed.

To detect a dropped request, the honest server needs to know how many messages are supposed to arrive in a given round. Related literature commonly assumes that protocol participation is static, i.e., that clients are always online [22], [23]. This is a very strong and arguably not very realistic assumption since it discards user churn. We introduce an additional mechanism that enables us to make weaker assumptions regarding client participation:

The *verifiable participation commitment* requires every client who joins the network to send a message to each server with which the client commits herself to participate in the next k rounds. The parameter k can be chosen by the client to fit his routine. A client could for example join when arriving at his office in the morning and commit to participating until his usual end of the workday. Clients can also commit to shorter periods and renew their commitment periodically. With that, the honest server knows from which clients to expect requests in any given round. If fewer requests than expected are received, the server assumes that a malicious drop must have occurred and refuses further participation to protect the senders' privacy. The adversary also needs to be prevented from replacing dropped requests with self-generated ones to circumvent the protection. This can be done by requiring the clients to include a *digital signature* with every request. That way, each request can be linked to the client who sent it and the server can *verify* that all committed clients have indeed participated.

B. Phase 2: Managing Published Messages.

When the writing epoch ends, the servers reveal the published messages among each other by combining their D_w states. As a first step, all cover messages (i.e., those which only contain zeroes) are discarded. Together, the servers choose a block size for D_r such that every topic's messages fit into a single block. Thus, the block size may be changing from round to round depending on the number of messages per topic, but at any point, all blocks of D_r have the same size. Next, the servers append each message in D_w to its corresponding topic-block in D_r. These messages are stored temporarily in D_r until the end of the communication round.

Updating the Topic List: Over time, new topics will be created and others will become inactive, thus there is a need for periodically updating the list of current topics and their corresponding blocks in D_r. To create a new topic, the client follows the same steps as when sending a message to an existing topic but includes a new topic id. The servers take notice of the unknown id and save the included message. During the next database update, a block for this new topic is created, and the topic is included in the list of topics sent to the clients. The first message from the original creator is added to this topic in the following round. Regarding deleting the inactive topics from D_r, servers will consider a topic as inactive, if there are no published messages on this topic for some configured number of rounds. The topic list and mapping from topic to block ID are updated periodically and clients are informed of the update afterward.

C. Phase 3: Anonymous Subscribing.

2PPS allows clients to anonymously subscribe to topics and get new messages without polling them. Like related protocols [9], [15], it depends on information-theoretic PIR (IT-PIR). The anonymous subscription consists of two building blocks: Subscription registration and message retrieval.

Private Subscription Registration: 2PPS requires all clients to update their subscriptions at a fixed rate to hide changes in interest. Every time a client receives a topic-list update, he has to renew his subscription. If the client is not interested in any topic, he sends a subscription request to a random topic. Clients that newly join the network need to wait until the next topic update to start participation.

Assume that Alice wants to subscribe to the jth topic. To do so, she creates a vector $q \in \{0,1\}^{\ell_r}$, which is equal to 1 at position j and equal to 0 at all other positions. Alice then computes a subscription request $req_i = Enc_{pk_i}(s_i|q_i)$ for each server S_i with $i \in \{1,\dots,N\}$. $Enk_{pk_i}(\cdot)$ is an encryption under the S_i's public key pk_i, $s_i \in \{0,1\}^{\ell_w}$ is a randomly chosen shared secret and the PIR query q_i is computed as follows:

$$q_i = \begin{cases} \text{random} & \text{for } i < N \\ q \oplus q_1 \oplus \cdots \oplus q_{K-1} & \text{for } i = N \end{cases}$$

The shared secret s_i is locally updated each round synchronously at client and server (e.g., using a cryptographic hash function or a key schedule).

To reduce the client's inbound bandwidth, related literature [9], [15] suggests the use of a random server P as a proxy for the client: Instead of sending the subscription requests $q_1,\dots,q_N$ directly to the servers, the client sends them to his proxy P. P

forwards these requests to corresponding servers where they are stored.

Remark 1 (Multiple Subscriptions): Each subscription registration may only contain a subscription to a single topic. If a client wants to be subscribed to multiple topics simultaneously, he has to send multiple subscription requests. To hide the number of topics clients are subscribed to, all clients need to send the same number of subscription requests. These can contain a mix of real subscriptions and cover subscriptions to random topics.

Private Messages Retrieval: In every round, each server S_i computes a response res_i for each stored subscription by taking the XOR of all D_r blocks that have 1 in their positions in the PIR query. Instead of sending the responses directly to the client, the servers submit them to P who computes $res \leftarrow \bigoplus_{i \in \{1,\ldots,N\}} res_i$, and forwards it to the client. Thus, the client's incoming bandwidth is reduced by a factor of N. To prevent P from learning which topic the client has subscribed to, each server has to obfuscate its response. Server S_i obfuscates its response by computing $res_i \leftarrow res_i \oplus s_i$, the client can restore the desired block of published messages by computing $res \oplus s_1 \oplus \cdots \oplus s_N$.

IV. Analysis of 2PPS security properties

In this section, we show that 2PPS reaches our formalized privacy goals as defined in Section II-C.

Theorem 1 (Publisher Unobservability): 2PPS achieves Publisher Unobservability.

Intuitively, reaching Publisher Unobservability requires unlinking senders from their messages and hiding which senders are active. To unlink senders from their messages, 2PPS employs secret sharing based on distributed point functions. Each server receives a secret share that does not reveal any information about the contained message by itself. Only once all clients have submitted their shares, the combination of all shares is revealed all at once. We strengthen the secret sharing scheme against adversaries in control of the whole network by introducing a timestamp and an additional layer of encryption around the shares. This prevents the adversary from being able to modify or replay shares. Further, we introduce the verifiable participation commitment which enables honest servers to detect dropped shares. Finally, we hide which senders are active by requiring all clients to send at a fixed rate, creating cover messages when they don't have a real message to send. A full proof of security can be found in Appendix A.

Theorem 2 (Subscriber Unobservability): 2PPS achieves Subscriber Unobservability.

In 2PPS, IT-PIR ensures that subscribers cannot be linked to the topics they're subscribed to. Both subscription requests and the responses containing the messages appear random to any adversary that is not in control of either the client itself or *all* servers. The use of cover traffic and synchronized round further ensures that the adversary cannot gain any information about the frequencies at which the client updates his subscription or receives messages. A full proof of security can be found in Appendix B.

V. Performance Evaluation

Privacy protection should not be only restricted to those with access to powerful hardware, but also it should support users who have limited bandwidth and computational power, e.g., in a mobile setting. 2PPS aims to keep the overhead at a reasonable level to enable as many clients as possible to participate. The goal of our evaluation is to investigate the impact of using DPFs (for private writing), and IT-PIR (for private reading) together on computation and network overhead on both client and server sides. One important measure of scalability is the end-to-end latency of a system. For 2PPS, we evaluate the influence of a changing number of participating clients, the number of subscribed topics per client, and the number of messages per topic on the latency of the system.

Implementation: A prototype of our protocol is implemented in C and Go. We use Go for the high-level operations of client and server. Cryptographic primitives are used from the DEDIS advanced crypto library and Go's native crypto library. We rely on the available source code of Express[1] for C implementations of the auditing protocol and DPFs. To update the shared secrets between clients and servers locally, we use keyed AES similar to Riffle [15]. We conduct the experiments on three virtual machines, each equipped with a 16-core Intel Xeon E5-2640 v2 processor and 64 GB of RAM. All three machines are located in the same data center. We operate two of them as servers and use the third one to simulate the clients. In all experiments, each client is configured to send one message per round (LS in Figure 2 refers to the length of the sent message). All clients participate in every round. Also, we test the performance of private retrieval for three different block sizes: 10 KB, 64 KB, and 256 KB (LR in Figure 2 refers to the length of the retrieved block). We adopt these block sizes from [23], [7] and [15].

Baselines: We compare 2PPS to three different PIR-based anonymous group communication protocols: Riposte [6], Pung [23], and Blinder [14]. We choose these protocols since they provide cryptographic anonymity guarantees similar to 2PPS. Riposte and Blinder support anonymous broadcasting, whereas Pung and 2PPS provide selective multicast communication (i.e., they allow the users to fetch only the messages that are interesting to them).

A. Computation Overhead

To understand the computation costs that are imposed on the client and server-side, we run a set of experiments in which every client sends one 1 KB message and retrieves one 64 KB block per round. Between experiments, we vary the number of messages processed by the servers (i.e., the number of participating clients). Since each client selects a random row to write its message into, collisions are possible and lead to the irreversible corruption of both colliding messages. In this experiment, we use a large fixed database D_w to handle this issue. This database achieves write success rates of 99.8 %, 98 %, and 82 % for 10^3 , 10^4 , and 10^5 messages respectively

[1] https://github.com/SabaEskandarian/Express

	# Messages Processed		
	10^3	10^4	10^5
Client CPU costs			
Generate DPF shares	229.26 μs	229.26 μs	229.26 μs
Audit	204.16 μs	204.16 μs	204.16 μs
Create PIR query	0.533 μs	6.73 μs	12.54 μs
Process PIR reply	19.33 μs	21.79 μs	22.027 μs
Server CPU costs			
Expand DPF shares	5.62 s	5.62 s	5.62 s
Audit	36.52 ms	36.52 ms	36.52 ms
Second Phase			
1. Combine D_w states	13.27 ms	14.01 ms	19.66 ms
2. Group messages	0.08 ms	0.87 ms	6.70 ms
Process PIR Query	0.17 ms	1.19 ms	10.06 ms

TABLE I: Cost of 2PPS operations under varying the number of messages stored on the server where the size of each message is 1 KB.

(according to the success rate formula in [6]). As the size of D_w is fixed, clients and servers pay fixed CPU costs for each write-request regardless of the number of messages received by the servers.

As shown in Table I, the client's operations are all comparatively inexpensive. Note that, "Create PIR query" is done only during subscription updates rather than every round. "Process PIR reply" denotes the time it takes to XOR the received PIR reply with the shared secrets to reveal the messages. The computation costs for the servers are dominated by the first phase ("Expand DPF shares" and the "Audit"). Also, the second phase introduces overhead which can be broken down further into the costs of two sub-phases: combining the shared D_w states, which accounts for the largest part of the overall time of the second phase, and the grouping of messages into their corresponding topics.

B. Network Overhead

In this section, we discuss the network overhead of 2PPS. These measures are especially important in bandwidth-restricted scenarios, such as mobile communication. We split our discussion into two parts: First, we consider the total bandwidth consumed by the operation of 2PPS versus related protocols. Second, we investigate how much "unnecessary" bandwidth in form of cover messages is required for 2PPS.

Comparative bandwidth comsumption: Figure 2a shows the total communication cost to send one message and retrieve one block by the client when the number of participating clients varies. That includes all the sent and received messages between the client and the server to achieve private writing and reading. Since we use a fixed size for D_w, the communication costs of private writing (including the auditing) are not growing when the number of clients increases.

Compared to Pung, 2PPS's anonymous writing is more expensive. However, Pung's reading phase introduces a high communication overhead that makes the total cost of communication using Pung significantly more expensive than 2PPS. For instance, when there are one million messages on the server, Pung has a total communication cost that is $1{,}263\times$ larger than 2PPS to send a 1 KB message and retrieve a 10 KB block. Note that, in the shown results, the costs of Pung include sending the PIR query by the client to retrieve the messages. If we assume that the PIR queries are stored already on the servers (similar to 2PPS), this will reduce the Pung costs to $950\times$ larger than 2PPS which is still a substantial difference. Pung's high overhead occurs because clients do not know the index of the data that they are interested in. Instead, they perform a binary search on the database through multiple CPIR queries. Further, the size of the CPIR answer increases as the CPIR recursion depth becomes higher. Angel et al. state that Pung's approach results in an increase of network overhead by a factor of $\log(n)$ for a database of n elements compared to PIR with known indices [23, Section 7.4].

Riposte also requires communication costs much higher than 2PPS. For 10^6 clients, 2PPS has $5{,}033\times$ less total cost than Riposte. 2PPS has better performance for two reasons: 1) it uses a new generation of DPFs [24] and an auditing protocol [7] that is more efficient than the one used in Riposte; 2) it sends to each client only a subset of the published messages, whereas Riposte broadcasts all messages to all clients.

Blinder introduces high total communication costs due to its need to operate by a large number of servers to achieve its anonymity guarantees (in our experiments, we ran Blinder on 5 servers). Additionally, it broadcasts all published messages to every client resulting in high communication overhead similar to Riposte. For 10^6 clients, the total network cost for each Blinder client is around 160 MB while the corresponding value in 2PPS is about 32 KB.

To further explore the performance implications of using IT-PIR on the network download overhead, Figure 2b depicts the total amount of data a client receives during 20 rounds for one subscription per client. In 2PPS, the amount of data per round equals the number of subscriptions a client has times the block size. A client who subscribes to one topic with a block size of 64 KB, downloads 1.28 MB during 20 rounds. Pung introduces a much higher communication overhead than 2PPS as well, but less than broadcasting. A Pung user downloads more data than a 2PPS user by a factor of $54-512\times$ to retrieve a 10 KB block.

To compare the download overhead in 2PPS with a broadcast-based approach such as Riposte and Blinder, we consider broadcasting under three different assumptions regarding the number of cover messages. The first case (denoted as "broadcasting 100 %") assumes that all messages that clients sent to the servers are real messages. Analogously, broadcasting 50 % and 25 % assume that 50 % and 25 % of messages are real, respectively. Note that cover messages are discarded and therefore not broadcast to the clients. For one million clients, the network download of broadcasting (100 %) is 20 GB in 20 rounds, resulting in a $15{,}625\times$ increase over 2PPS when the block size is 64 KB. Hence, this method is extremely inefficient in terms of bandwidth for popular services.

As shown in Figure 2c, 2PPS also considerably outperforms the three broadcasting variants when the number of subscriptions per client increases. Therefore, adopting IT-PIR in our

protocol to retrieve the interesting messages generally allows bandwidth-efficient communication between client and server. That makes 2PPS suitable for bandwidth-restricted users.

Average Cover Ratio: While 2PPS's retrieval method introduces no overhead for downloading a fixed amount of data, another kind of overhead occurs if topics have differing popularity: Similar to the "Counting Bound" proposed by Gelernter and Herzberg [25], hiding which topic a client is subscribed to *requires* a certain amount of overhead:

Assume that $\mathcal{A}$ knows the combined size of all messages sent to each topic and assume that each client subscribes to a single topic. To hide any information about the link between clients and their subscribed topics, *all* clients have to receive a response large enough that it could contain all messages from the most popular topic. Otherwise, $\mathcal{A}$ can eliminate topics from the list of possible subscriptions for a given client which have more messages than the client receives.

To fulfill the Receiver Counting Bound, 2PPS pads each topic to the same size as the most popular topic using cover massages; a client subscribed to any topic that receives fewer messages than the most popular one will receive some amount of cover messages with his PIR response creating network overhead. We want to evaluate how much of a client's request on average consists of cover messages. While we have no real usage data for 2PPS, we can find related literature to approximate how topic popularity might be distributed: According to Chen et al. [26], the popularity distribution of Twitter hashtags follows Zipf's law with an exponent of $\alpha \approx 0.8$. Further, Liu et al. [27] have analyzed the RSS feed characteristics and determined that the popularity of feeds sorted by the number of requests also follows Zipf's law with an exponent of $\alpha \approx 1.37$. We use Chen et al.'s result to approximate the number of messages sent to a given topic and Liu et al.'s result to determine the likelihood of a client subscribing to a given topic. In our experiment, we assumed that the most popular topic receives 1000 messages and that the ith most popular topic receives $1/i^{0.8}$ as many messages. We chose a random topic index j according to a Zipfian distribution with parameter $\alpha = 1.37$ and determined how many cover messages topic j required. Under these assumptions, clients receive approximately 54 % cover messages per retrieval on average. Note that this number highly depends on the actual usage patterns of the service: If all topics receive the same number of messages, *no* cover messages are required, if topic popularity varies widely, more than 54 % cover messages might be received on average.

C. End-to-End Latency

To evaluate the efficiency of our protocol, we are interested in computing the total time required to send one message and retrieve one block by each client. The latency in 2PPS is measured as the total time of all three protocol phases. The time of DPF evaluation represents the expensive part of the total latency, and it depends on the size of D_w. Having a fixed large database means a fixed big evaluation time for DPF shares even when the number of received write requests is small. For less latency, the evaluation time can be reduced by changing the size of D_w based on the number of expected requests. However, the database should be still large enough to handle requests successfully with high probability.

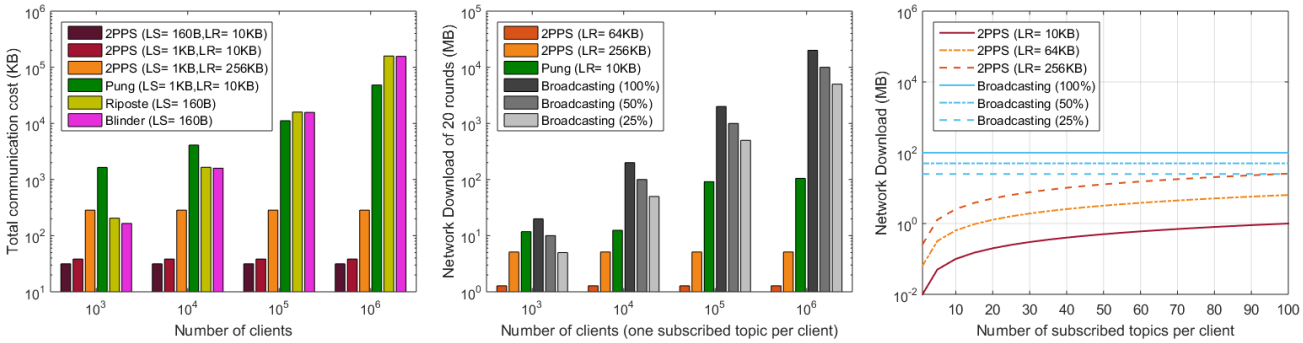
The latency of the third phase is determined by the number of topics, the block size, and the number of subscriptions per client. The more topics the read database contains or the larger each topic block is, the longer it takes to compute a PIR reply. If a client subscribes to more topics, the number of PIR replies that need to be computed increases, increasing latency.

Figure 2d illustrates how the retrieval time of one block scales whith varying block sizes and numbers of clients. In general, retrieving messages using either our method or broadcasting doesn't cause much latency. We compared our protocol to broadcasting to understand the performance implication of using PIR for distributing messages to the receivers instead of using a broadcast. 2PPS is considerably faster than Pung, even when retrieving larger blocks.

Figure 2e illustrates the increase in retrieval time when a 2PPS client subscribes to many topics. The number of participating clients influences the time required to broadcast messages more than it does with 2PPS. That is especially true when most of the published messages are real. For 10^4 clients, retrieving 12 blocks using 2PPS takes 97 ms more than the broadcasting, which means the difference in the latency time between the two approaches is comparatively insignificant. Therefore, 2PPS's retrieval method can reduce the bandwidth without leading in return to high latency, even if the clients have many subscriptions.

Figure 2f shows the end-to-end latency for posting one message when we vary the number of clients. Again, this figure demonstrates the negligible effect of retrieval time on the total latency, as the end-to-end latency of 2PPS is slightly smaller than the method that relies on DPF and broadcasting (instead of IT-PIR). 2PPS has significantly better performance than Riposte as it adopts a more efficient DPF version [24] and auditing method [7].

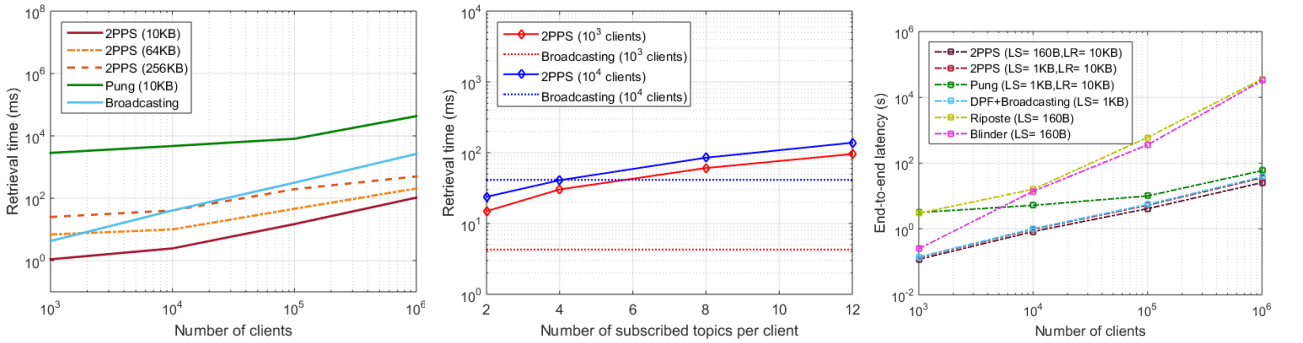
We compared the latency of 2PPS to Pung when sending one 1 KB message and retrieving one 10 KB block (10 messages in the block). As shown in Figure 2f, 2PPS outperforms Pung especially for a modest number of clients. In Pung, communication partners are required to trust each other, which enables clients to agree upon secret mailboxes for their message exchange. Since the link between clients and their mailboxes is not known to the adversary, messages can be directly written to the mailboxes, resulting in much less overhead than our DPF-based approach. The use of a single untrusted server in Pung requires much more overhead in the retrieval of messages than IT-PIR based approach of 2PPS. Overall, this expensive reading phase more than offsets any advantages in the latency of Pung's writing phase. The latency of Blinder is close to Riposte and considerably higher than 2PPS. Similar to 2PPS, the most expensive part in Blinder is the process of expanding the submitted blind write requests to add them to the database. For 10^5 clients, the total time to serve this number of clients is around 8 minutes, and the expanding process acquires 96 %



(a) Total communication costs when the client sends one message and retrieves one block of messages.

(b) The total amount of downloaded data by a client after 20 rounds with varying numbers of clients and block sizes.

(c) The total amount of downloaded data by a client for 100,000 clients with varying numbers of subscribed topics and block sizes.



(d) Retrieval time of one block per client with varying number of clients and block size.

(e) Retrieval time when a client subscribes to different number of topics.

(f) End-to-end latency of message delivery when a client subscribes to one topic.

Fig. 2: Evaluation Results

of the total latency. 2PPS supports the same number of clients with less than half of Blinder's time.

VI. Related Work

2PPS provides a publish/subscribe protocol that achieves provable publisher and subscriber unobservability against a global active adversary who may also corrupt arbitrary clients and all but one server. Privacy protection is achieved via a combination of DPF-based secret sharing for publishing and PIR for subscribing to topics. In this section, we provide an overview of existing anonymous communication protocols and how they compare to 2PPS.

Mixnets-based Protocols.: Mix networks (mix nets) [28]–[30] ensure the anonymity of users by obfuscating the source of a message. They work by collecting the messages from many users and shuffle them by a set of servers called mixes before sending them out to recipients. Therefore, they make it difficult for the global adversary to correlate input and output messages and protect against traffic analysis attacks. However, malicious mixes can launch several attacks to deanonymize users; for instance, they can drop, modify, or duplicate the input messages before sending them out [31]–[33]. Verifiable shuffles techniques [34]–[36] have been proposed to protect against tampering messages by malicious servers, but these techniques introduce high computation overhead. Mixnets-based protocols like McMix [37], Atom [38] and XRD [39] induce high latency to support large numbers of users. While protocols as Vuvuzela [22], Stadium [40], Alpenhorn [41], and Karaoke [42] achieve better performance than 2PPS, but they provide weaker differential privacy guarantees compared to the cryptographic guarantees as 2PPS does. In practice, this means that an adversary learns more information the longer he observes the mixnet, which is not the case for 2PPS.

DCnets-based Protocols.: Chaum's Dining Cryptographers network (DCnet) [43], [44] is an information-theoretic anonymous broadcast method. To increase the scalability and practicality of DCnets, many protocols like Dissent [8] and Verdict [45] adopted the client-server paradigm, where n clients form the anonymity set, but only a small set of N servers implement the functionality. This adoption reduces the overall communication complexity from $O(n^2)$ in the traditional DCnet model (where there is a full graph between clients) to $O(N \cdot n)$ in the client-server DCnet model. The sender anonymity guarantees that DCnet protocols provide are the same as 2PPS does. However, the size of the sent and the received message in 2PPS is significantly smaller than in DCnet protocols. That is because 2PPS uses DPF to compress the write requests and IT-PIR to only retrieve the blocks that the client is interested in,

instead of getting all the messages through broadcasting. Also, the used primitives in our protocol allow it to support a much larger number of clients and provide faster communication time than DCnet protocols do [6], [7].

PIR-based Protocols.: Many protocols rely on PIR methods to enable anonymous communication. There are two classes of these protocols that are: 1) information theoretic-PIR-based protocols (multiserver) such as Express [7], Riposte [6], Blinder [14], and Talek [9]; and 2) computational-PIR-based protocols (one server) such as Pung [23]. Express uses DPF to allow a user to send a message anonymously to the mailbox of another user. However, this protocol does not protect the anonymity of the mailbox's owner (i.e., the receiver's anonymity). Riposte and Blinder also use DPF but broadcast all the published messages. As we have shown in our evaluation (Section V), broadcasting results in much higher network overhead than 2PPS's PIR, making it not suitable for bandwidth-restricted scenarios. DPF-based protocols in general have a comparatively low computational overhead and can scale to support millions of users. Talek provides a private publish-subscribe protocol that allows communication between small groups of trusted users. In contrast, 2PPS's overhead does not depend on the number of subscribers a topic has and users neither have to trust publishers nor other subscribers to protect their privacy. Pung operates in a single-server setting and provides strong anonymity guarantees since it can hide user's interests even if the server is malicious. However, it introduces more overhead than 2PPS and requires users to trust their communication partners.

VII. Discussion

In this section, we present some lessons learned from designing a public-message protocol based on secret sharing and private information retrieval.

Availability: As discussed in Section III, 2PPS owes its strong provable privacy protection to the combination of secret sharing and private information retrieval. Both of these techniques distribute trust by relying on multiple servers. This distribution of trust is great from a privacy perspective: As long as one server remains honest, user privacy is preserved. Users with high privacy requirements can even deploy their own servers to increase the chance of an honest one existing.

While very advantageous when it comes to privacy, secret sharing and PIR are very vulnerable as far as availability is concerned: If a *single* malicious server refuses to provide its write-database state after clients sent their messages, the protocol execution cannot continue. To reveal the messages, *all* shares are required by design. The same problem arises on the reading site: If a malicious server refuses to send his PIR response, the clients will not be able to receive the message from their subscribed topics. Thus, as related literature [6], [9], we assume that malicious servers do not target availability.

Remark 2 (Backup Servers): An obvious mitigation against availability attacks from servers would be to introduce a "backup" server for each server. The backup server would receive the same information as its corresponding main server. In case the main server refuses to submit his shares, the backup server can step in. This avoids disruption as long as not both one main server and its backup refuse to participate. However, this comes at the cost of additional trust. Instead of only requiring one server to be honest, two honest servers are required; One honest main server and one honest backup server. Further, communication overhead for publishing messages also increases twofold, since the number of servers is doubled.

Malicious clients may also target availability. While the auditing protocol prevents clients from submitting malformed requests that would corrupt the write database, there are other avenues of attack which are inherent to 2PPS' open nature: An adversary could create a large number of topics, increasing the required size of the write- and read databases and therefore also the network overhead of the whole system. Further, the adversary could also spawn a large number of clients who all submit messages to a single topic, increasing the amount of cover needed for all other topics. In scenarios where availability is of greater concern than the open nature of the systems, mechanisms can be put in place that increase the effort of adding users and topics to the system.

Mitigating Intersection Attacks: Due to the public nature of messages and topics, 2PPS is particularly vulnerable to a specific kind of intersection attack: Assume that the adversary $\mathcal{A}$ wants to find out the interests of client Alice, who only publishes to a single topic. Every time Alice is participating in a communication round, $\mathcal{A}$ records which topics are active (i.e., have messages sent to them). Over multiple rounds, $\mathcal{A}$ *intersects* the list of active topics until only a single topic remains, which is unambiguously linked to Alice.

Intersection attacks are inherently possible in any protocol that allows users to choose when they want to participate (see Appendix C for proof). While we assume constant participation for our security analysis, we also present possible mitigation techniques against intersection attacks for situations where constant participation is not obtainable:

- **Cover Traffic.** A simple mitigation that is already in use with 2PPS is cover traffic. If Alice uses cover traffic, then $\mathcal{A}$ cannot distinguish rounds where Alice is sending "real" messages from rounds where she is participating idly, increasing the number of rounds $\mathcal{A}$ needs to observe.
- **Delayed Publishing.** Alice can also require that her message is not published in the round where she sent it but later (when Alice might already be offline). To do so, Alice can include some random delay d in the message-topic tuple $t \mid m$ and *encrypt* it with the public key of one of the servers. After combining their D_w states, the corresponding server will reveal the ciphertext and delay publishing it. This solution is based on the idea of distributing knowledge which means each server knows the real publishing time of only a subset of all messages. Thus, $\mathcal{A}$ who controls one of the servers cannot accurately link every message to the set of the potential senders.

Collisions: In 2PPS, the client chooses a random row in the database to write her message into. Therefore, it is possible to have collisions, i.e., two or more clients writing their messages

in the same row which corrupts both messages. To minimize the probability of collisions during normal operation, 2PPS uses a write database that is much larger than the number of participating clients. However, this solution cannot solve the problem completely. If a client does not find her message among the published message in a given round, she has to assume that a collision occurred and may try to send her message again in a later round. Blinder [14] employs another approach to deal with collisions: If a client wants to publish message m, she computes one set of secret shares with m at a random index and another set of shares with the square of m at the same index. The servers store the received shares in separate databases. If a single collision occurred for a given index, the combined shares of both databases form a solvable system of quadratic equations, enabling the servers to reconstruct both original messages. While this reduces the required storage space on the server side, it requires more computational overhead for clients and servers.

VIII. Conclusion & Future Work

2PPS provides an anonymous publish/subscribe protocol with strong provable privacy guarantees for both publishers of messages and subscribers. This is achieved by combining secret sharing based on distributed point functions for publishing with private information retrieval for accessing subscribed topics. Compared to previous work, additional protection mechanisms are introduced to the secret sharing: A combination of timestamps and encryption allows 2PPS to provide publisher privacy not only against a malicious server but also against stronger adversaries that can observe and interfere with traffic on all network links. Our experimental evaluation shows that 2PPS can support a large number of users with latency suitable for applications such as microblogging and newsfeeds. While 2PPS reaches its stated goals, we see the following avenues for future improvement:

- Improvements to the auditing protocol to increase the number of servers without introducing high computation and communication costs.
- Enabling users to subscribe to multiple topics using one subscription request. If servers would be able to handle all the subscriptions of one client at once, more efficient packing of messages may be possible, reducing the amount of cover needed.
- Allowing the anonymous retrieval of messages from previous rounds.
- Lowering latency to enable use cases such as live streaming.

Acknowledgments

We thank Martin Byrenheid, Clemens Deusser, and Ephraim Zimmer for their valuable feedback and discussion.

References

[1] A. Hern. Firechat updates as 40,000 iraqis download 'mesh' chat app in censored baghdad. *The Guardian*, 2014.
[2] A. Bland. Firechat – the messaging app that's powering the hong kong protests. *The Guardian*, 2014.
[3] A. Herasimenka et al. There's more to belarus's 'telegram revolution' than a cellphone app. *The Washington Post*, 2020.
[4] C. Baraniuk. Firechat warns iraqis that messaging app won't protect privacy. *Wired*, 2014.
[5] M. Smithberger et al. Making sense of the $1.25 trillion national security state budget. *POGO.org*, 2019.
[6] H. Corrigan-Gibbs et al. Riposte: An anonymous messaging system handling millions of users. In *IEEE S&P*, 2015.
[7] S. Eskandarian et al. Express: Lowering the cost of metadata-hiding communication with cryptographic privacy. *ArXiv*, 2019.
[8] D. Wolinsky et al. Dissent in numbers: Making strong anonymity scale. In *USENIX OSDI*, 2012.
[9] R. Cheng et al. Talek: a private publish-subscribe protocol. Technical report, 2020.
[10] G. Perng et al. M2: Multicasting mixes for efficient and anonymous communication. In *ICDCS*, 2006.
[11] S. Arnautov et al. Pubsub-sgx: Exploiting trusted execution environments for privacy-preserving publish/subscribe systems. In *SRDS*, 2018.
[12] G. Giakkoupis et al. Privacy-conscious information diffusion in social networks. In *DISC*, 2015.
[13] D. Lin et al. Scalable and anonymous group communication with mtor. *PETS*, 2016.
[14] I. Abraham et al. Blinder: Mpc based scalable and robust anonymous committed broadcast. *IACR Cryptol. ePrint Arch.*, 2020.
[15] A. Kwon et al. Riffle: An efficient communication system with strong anonymity. *PETS*, 2016.
[16] C. Kuhn et al. Sok on performance bounds in anonymous communication. In *WPES*, 2020.
[17] D. Wolinsky et al. Dissent in numbers: Making strong anonymity scale. In *USENIX OSDI*, 2012.
[18] C. Kuhn et al. On privacy notions in anonymous communication. *PETS*, 2019.
[19] A. Shamir. How to share a secret. *Commun. ACM*, 1979.
[20] E. Boyle et al. Function secret sharing: Improvements and extensions. *SIGSAC*, 2016.
[21] M. Ando et al. On the complexity of anonymous communication through public networks. *ArXiv*, abs/1902.06306, 2019.
[22] J. Van Den Hooff et al. Vuvuzela: Scalable private messaging resistant to traffic analysis. In *SOSP*, 2015.
[23] S. Angel et al. Unobservable communication over fully untrusted infrastructure. In *USENIX OSDI*, 2016.
[24] E. Boyle et al. Function secret sharing: Improvements and extensions. In *SIGSAC*, 2016.
[25] N. Gelernter et al. On the limits of provable anonymity. In *WPES*, 2013.
[26] H. Chen et al. Scaling laws and dynamics of hashtags on twitter. *Chaos*, 2020.
[27] H. Liu et al. Client behavior and feed characteristics of rss, a publish-subscribe system for web micronews. In *IMC*, 2005.
[28] D. Chaum et al. cmix: Anonymization by high-performance scalable mixing. In *ACNS*, 2017.
[29] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 1981.
[30] A. Jerichow et al. Real-time mixes: A bandwidth-efficient anonymity protocol. *IEEE Journal on Selected Areas in Communications*, 1998.
[31] B. Pfitzmann et al. How to break the direct rsa-implementation of mixes. In *EUROCRYPT*, 1989.
[32] L. Nguyen et al. Breaking and mending resilient mix-nets. In *PETS*, 2003.
[33] B. Pfitzmann. Breaking an efficient anonymous channel. In *EUROCRYPT*, 1994.
[34] S. Bayer et al. Efficient zero-knowledge argument for correctness of a shuffle. In *EUROCRYPT*, 2012.
[35] J. Furukawa et al. An efficient scheme for proving a shuffle. In *CRYPTO*, 2001.
[36] J. Brickell et al. Efficient anonymity-preserving data collection. In *SIGKDD*, 2006.
[37] N. Alexopoulos et al. Mcmix: Anonymous messaging via secure multiparty computation. In *USENIX Security*, 2017.
[38] A. Kwon et al. Atom: Horizontally scaling strong anonymity. In *SOSP*, 2017.
[39] A. Kwon et al. Xrd: Scalable messaging system with cryptographic privacy. In *USENIX NSDI*, 2020.
[40] N. Tyagi et al. Stadium: A distributed metadata-private messaging system. In *SOSP*, 2017.

[41] D. Lazar et al. Alpenhorn: Bootstrapping secure communication without leaking metadata. In *USENIX OSDI*, 2016.
[42] D. Lazar et al. Karaoke: Distributed private messaging immune to passive traffic analysis. In *USENIX OSDI*, 2018.
[43] D. Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of cryptology*, 1988.
[44] P. Golle et al. Dining cryptographers revisited. In *EUROCRYPT*, 2004.
[45] H. Corrigan-Gibbs et al. Proactively accountable anonymous messaging in verdict. In *USENIX Security*, 2013.
[46] D. Wolinsky et al. Hang with your buddies to resist intersection attacks. *SIGSAC*, 2013.

APPENDIX

A. Proving Sender Unobservability

Lemma 1 (Message Unlinkability): $\mathcal{A}$ cannot identify which sender sent a given message.

Proof: We proof Lemma 1 by showing that $\mathcal{A}$ cannot use any of his abilities to do so.

- **Passive Observation.** Every client sends one request per round to each server. All requests are encrypted under the receiving servers public key. The servers collect the incoming requests and reveal all messages from the current round at once. Thus, $\mathcal{A}$ cannot link messages to senders by passively observing requests between clients and servers.
- **Server Corruption.** According to our assumptions, $\mathcal{A}$ is able to corrupt all but one 2PPS servers. $\mathcal{A}$ can only match a client to the request he sends prior to adding it to the db_w state. However, at this point, $\mathcal{A}$ can learn at most $N-1$ of the clients N DPF-shares. Related literature has formal proves that combining $N-1$ shares does not reveal any information about the enclosed message [24].
- **Replay.** $\mathcal{A}$ could either replay request during the same or a later round. If $\mathcal{A}$ replays a request during the same round as the original request was sent, the honest server will detect identical requests arriving and discard all but one. If $\mathcal{A}$ replays a request during some later round, the honest server will notice that the included timestamp is not valid and discard the request. $\mathcal{A}$ is not able to update the timestamp of the replayed request, since it is protected by an encryption layer prior to the honest server.
- **Modification.** To be able to link a request to the client who sent it, $\mathcal{A}$ has to modify the request prior to the honest server. Since the contained DPF-share is encrypted, any modification at this point will lead to unpredictable changes of the share. Such a share will be rejected by the honest server's auditing protocol with overwhelming probability.
- **Dropping.** We assume that the honest server can detect a dropped request and refuse further protocol participation.

■

Theorem 3 (Sender Unobservability): 2PPS achieves Sender Unobservability.

Proof: We define a series of hybrid games:

- H_0: The original $S\overline{O}$ game
- H_1: H_0, but clients can only publish 0-messages to a random topic
- H_2: H_1, but clients can only publish cover messages
- H_3: Identical Scenarios

In the following, we show that any adversary that can win H_i non-negligible advantage can also win H_{i+1} with non-negligible advantage.

- $H_0 \approx H_1$: We assume that $\mathcal{A}$ can win H_0. Lemma 1 states that $\mathcal{A}$ does so without being able to link any revealed message to a sender.
- $H_1 \approx H_2$: The only difference between a cover message and a "real" message is it's content: While a real message is sent to a chosen topic and can contain an arbitrary plaintext, a cover message is sent to a random topic and contains only 0s. We note that H_1 already requires real messages to have identical content to cover messages. Thus, all messages in H_1 are indistinguishable from cover messages to $\mathcal{A}$. If $\mathcal{A}$ can win H_1, he can therefore also win H_2.
- $H_2 \approx H_3$: As mentioned previously, we assume that all clients participate in every round. Further, every client sends exactly one message in each round. In H_2, all messages are sent to a random topic and contain only 0. Thus, $\mathcal{A}$ already has no influence on protocol activity in H_2 and the scenarios therefore appear identical to him.

We have shown that any adversary who can win H_0 can also win H_3 with a non-negligible advantage. Since $\mathcal{A}$ is only allowed to submit identical scenarios in H_3, there cannot be any such $\mathcal{A}$. Therefore, no $\mathcal{A}$ can win H_0, which is equivalent to the $S\overline{O}$ game. ■

B. Proving Receiver Unobservability

Theorem 4 (Receiver Unobservability): 2PPS achieves Receiver Unobservability.

Proof: With Receiver Unobservability, $\mathcal{A}$ may not learn any information about receiver activity. $\mathcal{A}$ can either gain information about receivers by observing subscription registrations or communication.

- **Subscription Registration.** Each client sends his subscription registrations at a fixed rate to each server. Prior to the receiving server, the registration is always encrypted under the public key of this server. An active adversary may be able to corrupt a clients subscription registration. However, this does not lead to differing behavior based on the topic the client wants to subscribe to, since receivers show do not react to the messages they receive.
  We assume that $\mathcal{A}$ can corrupt $N-1$ servers. This does not help him in determining the topic a client is subscribing to, since any combination of $N-1$ requests per definition appears random to $\mathcal{A}$. Only the combination of all N requests reveals the topic.
- **Communication.** Since the subscription request appears random to every server, no combination of $N-1$ servers can determine which topic the client is subscribed to by computing the response res_i. Although the primary server P has access to all individual responses, it cannot reveal the messages, since they are obfuscated by the shared secret s_j between the client and the honest server.

■

C. Intersection Attacks

Definition 2 (Delivery-Guarantee): A protocol provides Delivery-Guarantee, if all messages that are sent in a given round are also published in the same round.

Definition 3 (Sending-Nonobligation): A protocol provides Sending-Nonobligation, if it does not enforce when users send messages.

Common approaches against intersection attacks [46] do *not* provide Sending-Nonobligation.

Theorem 5: A protocol that provides both Delivery-Guarantee and Sending-Nonobligation cannot provide sender-messages unlinkability against an adversary who learns which messages are sent if the protocol guarantees delivery of all messages sent in a given round.

Proof: Without Sending-Nonobligation, clients may only participate in a subset of all rounds. $\mathcal{A}$ observes a messages m being published in round r Due to Delivery-Guarantee, a client that has not participated in round r cannot have been the sender of m.

If $\mathcal{A}$ can determine that another message m' was sent by the same sender, he can further narrow down the set of possible senders of m and m' via an *intersection* attack. ■

P2

ANONIFY: DECENTRALIZED DUAL-LEVEL ANONYMITY FOR MEDICAL DATA DONATION

This chapter was first published as

Sarah Abdelwahab Gaballah, Lamya Abdullah, Mina Alishahi, Thanh Hoang Long Nguyen, Ephraim Zimmer, Max Mühlhäuser, and Karola Marky. "Anonify: Decentralized Dual-level Anonymity for Medical Data Donation." Proceedings on Privacy Enhancing Technologies 3 (2024): 94-108.

An artifact for this work was approved as "Artifact Reproduced" by PETS, and it is available at: https://github.com/lng-ng/anonify

Contribution Statement: I led the idea generation, conceptualization, and development of the proposed work, as well as the experiment design, data analysis, and writing of the publication. All co-authors helped with critiques and comments on the concept design and participated in the creation of the publication.

Anonify: Decentralized Dual-level Anonymity for Medical Data Donation

Sarah Abdelwahab Gaballah
Ruhr University Bochum
sarah.gaballah@rub.de

Lamya Abdullah
Technical University of Darmstadt
abdullah@tk.tu-darmstadt.de

Mina Alishahi
Open Universiteit
mina.sheikhalishahi@ou.nl

Thanh Hoang Long Nguyen
Technical University of Darmstadt
long.nguyen@stud.tu-darmstadt.de

Ephraim Zimmer
Technical University of Darmstadt
zimmer@privacy-trust.tu-darmstadt.de

Max Mühlhäuser
Technical University of Darmstadt
max@tk.tu-darmstadt.de

Karola Marky
Ruhr University Bochum
karola.marky@rub.de

ABSTRACT

Medical data donation involves voluntarily sharing medical data with research institutions, which is crucial for advancing healthcare research. However, the sensitive nature of medical data poses privacy and security challenges. The primary concern is the risk of de-anonymization, where users can be linked to their donated data through background knowledge or communication metadata. In this paper, we introduce *Anonify*, a decentralized anonymity protocol offering strong user protection during data donation without reliance on a single entity. It achieves dual-level anonymity protection, covering both communication and data aspects by leveraging Distributed Point Functions, and incorporating k-anonymity and stratified sampling within a secret-sharing-based setting. *Anonify* ensures that the donated data is in a form that affords flexibility for researchers in their analyses. Our evaluation demonstrates the efficiency of *Anonify* in preserving privacy and optimizing data utility. Furthermore, the performance of machine learning algorithms on the anonymized datasets generated by the protocol shows high accuracy and precision.

KEYWORDS

Medical Data Donation, Data Anonymity, Anonymous Communication, Distributed Point Functions, k-anonymity, Stratified Sampling

Proceedings on Privacy Enhancing Technologies 2024(3), 94–108

https://doi.org/10.56553/popets-2024-0069

1 INTRODUCTION

Medical data donation[1] is a voluntary act where individuals share their health-related information with researchers to support scientific research, medical advancements, and public health initiatives [4]. However, medical data donation faces significant challenges primarily due to privacy and security concerns based on the sensitivity of medical data. The most critical concern revolves around the potential for the risk of individuals being identifiable through their data [35, 40]. Therefore, it is crucial to provide strong protection guarantees for users when they donate their medical data.

Medical data is typically provided to researchers in the form of a relational (tabular) structure. A table is composed of columns (attributes) and rows (records), with attributes categorized as *direct identifiers*, *quasi-identifiers* (QIDs), *sensitive attributes* (SAs), or *non-sensitive attributes* [26]. Direct identifiers, such as names or social security numbers, explicitly identify record owners. QIDs, like age, job, sex, or zip code, may not identify individuals on their own but could if combined. SAs encompass sensitive person-specific information, such as diseases. Non-sensitive attributes include those that do not fit into the above categories.

Simply removing direct identifiers from data is not sufficient to prevent re-identification [38]. It has been shown that if an individual's record is unique based on QIDs, an attacker with this information can directly link the record to its owner, leading to *identity disclosure*.

Even in cases where a group of individuals in a dataset shares identical QID values, the absence of diversity in the SA values within the records of these individuals can make them vulnerable to *attribute disclosure attacks* [26]. In Table 1, we present an example of such attacks, illustrating two groups where individuals in each exhibit similarity in QID values. Specifically, the first three records belong to one group, while the remaining records belong to another group. An adversary can deduce that any individual with a record in the first group has hepatitis, as all records in the group share

[1] In medical data donation scenarios, data is gathered by research institutes from individuals through apps, such as the Corona-Datenspende app [30]. The set of data donors may include both those currently experiencing health issues and those in good health. This differs from traditional medical data collection scenarios, where data is usually collected by hospitals or medical institutions as part of the treatment process.

Table 1: Example for the attribute disclosure attacks.

No.	Age	Sex	Zip Code	Disease
1	43	Male	56126	Hepatitis
2	43	Male	56126	Hepatitis
3	43	Male	56126	Hepatitis
4	35	Female	56121	Coronary Heart Disease
5	35	Female	56121	Arrhythmia
6	35	Female	56121	Valve Disease

identical SA values. Similarly, each individual in the second group can be inferred to have a heart-related disease due to SA values in the records that imply a shared trait.

To mitigate de-anonymization, various techniques have been introduced, with the most popular ones being k-anonymity for addressing identity disclosure attacks, and ℓ-diversity and t-closeness for mitigating attribute disclosure attacks. These techniques are designed for a centralized setting where a single entity is responsible for aggregating all individuals' medical data [6]. This implies that the entity has knowledge about each individual and their corresponding medical information, requiring users to place trust in this entity. Such a requirement may potentially reduce users' willingness to donate their medical data. Furthermore, the centralized nature of this entity introduces a single point of failure. In the event of a data breach, the privacy of all users could be compromised.

This paper addresses de-anonymization attacks, specifically identity and attribute disclosure, and issues arising from centralized medical data sharing. It proposes *Anonify*, a decentralized anonymity protocol designed for medical data donation. It offers dual-level anonymity protection: (1) at the communication and (2) at the data level. *Anonify* guarantees anonymous communication to prevent any linkability between users and their communicated records, enabling users to donate data without the need to place their trust in a single entity. This protection is achieved through a secret-sharing-based method for anonymous writing called *Distributed Point Functions* (DPF)[7], coupled with a broadcasting-based approach for anonymous data retrieval. Additionally, to defend against de-anonymization risks associated with donated medical data, it employs k-anonymity[38] and *stratified sampling* [28], all within decentralized settings.

Anonify consists of two phases: the registration phase and the publishing phase. In the registration phase, users employ DPF to anonymously submit records containing their QID values to an aggregator operated by multiple collaborating servers. To guard against identity disclosure attacks, *Anonify* applies k-anonymization to these records and organizes them into groups based on QID similarity. Using a broadcast-based approach, users can then anonymously learn their corresponding group. In the second phase, using DPF, users anonymously transmit their medical data (SA values) associated with their group identifier to the aggregator. To protect against attribute disclosure attacks, *Anonify* conducts stratified sampling on encrypted data, revealing only a portion of records in each group. Finally, the protocol disseminates the anonymized sampled donated data to researchers.

Contributions: In this paper, we make the following contributions:

- We introduce *Anonify*, a decentralized protocol for anonymous medical data donation, ensuring protection at communication and data levels. By leveraging DPF, *Anonify* aggregates data from users and applies k-anonymity and stratified sampling without relying on a single entity. This prevents the aggregator from de-anonymizing users, even with communication metadata or identity and attribute disclosure attacks. Additionally, our novel application of DPF facilitates the employment of stratified sampling without requiring trust in the aggregator with the entire set of records.
- We conduct a security analysis to demonstrate that *Anonify* achieves our security goals in terms of anonymous communication and data anonymity.
- We assess the efficiency of *Anonify*, utilizing a realistic medical dataset to simulate user-submitted records. Our evaluation incorporates multiple utility and privacy metrics, along with an examination of the data distribution characteristics post-application of *Anonify*. To ensure the anonymized data allows accurate analysis, we tested eight well-known machine learning classifiers on the anonymized dataset, revealing results closely resembling those of the original non-anonymized dataset.

The remainder of this paper is organized as follows: Section 2 provides the necessary background knowledge about anonymous communication and data anonymity. In Section 3, we explain our system and threat model, along with the security properties provided by our protocol. Section 4 introduces our protocol for decentralized anonymous medical data donation. In Section 6, we describe the evaluation of our approach. Section 7 presents related work, and in Section 8, we conclude our paper.

2 BACKGROUND

In this section, we explain the methods we use in our protocol, namely DPF, k-anonymity, generalization, and sampling.

2.1 Distributed Point Functions

Distributed Point Functions (DPF) [7, 13] are cryptographic constructs designed to facilitate secure and privacy-preserving computations in distributed or decentralized environments. DPF enables users to write in a database D distributed across a set of servers S without any of the servers being able to link any user to the specific message they wrote. This guarantee is achieved if at least one of the servers is honest.

To describe how anonymous writing can be done using DPF, we first introduce a basic secret-sharing-based approach, then explain how DPF improves this approach to enable efficient anonymous writing.

Suppose there are n servers, with each server storing a full copy of a database D. All servers collectively maintain the contents of this database. A user aims to submit a message m_i to D without the n servers storing D being able to link the message to the user's identity. The naive approach to achieve this is as follows:

Initially, the user computes a vector v with the same length as the database D. This vector contains the message m at a randomly selected index t (chosen locally by the user) and 0 at all other

indices. Subsequently, the user generates n secret shares $v_1, \ldots, v_n$ that satisfy the following properties:

(1) $\sum_{i=1}^{n} v_i = v$
(2) Any combination of $n-1$ secret shares does not reveal information about m or the index where m is located.

The user then distributes these shares to the servers, with the i-the server, $s_i \in S$, receiving v_i. Each s_i adds v_i to its database instance D^i using the operation $D^i \leftarrow D^i + v_i$.

After processing requests from multiple users, the servers collaborate to compute a combined database $D = \sum_{i=1}^{n} D^i$. Assuming each user selected a unique index for their messages, D contains all original messages.

For this method to work, transmitting a vector with the same size as the database for each write request is needed, making this method inefficient. To address this inefficiency, Corrigan-Gibbs et al. [7] proposed DPF to *compress* the shares transmitted to the servers.

Definition 1 (DPF). *Let $f_{t,m} : \{0, \ldots, \ell\} \mapsto \mathbb{F}$ be a point function with*

$$f_{t,m}(j) = \begin{cases} m & \text{for } j = t \\ 0 & \text{for } j \in \{0, \ldots, \ell\} \setminus t \end{cases}$$

$f_A, f_B : \{0, \ldots, \ell\} \mapsto \mathbb{F}$ are distributed point functions of $f_{t,m}$ if:

(1) neither f_A nor f_B individually reveal any information about m or t, and
(2) $\forall\, j \in \{0, \ldots, \ell\} : f_A(j) + f_B(j) = f_{t,m}(j)$

To generate n DPF-shares $f_1, \ldots, f_n$ containing m at a random index t, the user employs $\mathsf{GenDPF}(m, t)$. These shares are distributed among the servers, with each server s_i receiving f_i. The server s_i can derive v_i by evaluating $f_i(j)$ at every point $j \in \{0, \ldots, \ell\}$. Current research indicates that sending a DPF share instead of v_i reduces the communication cost to $O(\lambda \cdot \log \ell + |m|)$ bits, where λ is a security parameter [5].

2.2 k-anonymity

To mitigate identity disclosure attacks, it is crucial to anonymize the QID values in datasets. A widely used concept for this is *k-anonymity* [38], which modifies the dataset to ensures that at least k records in the dataset share the same QID values. k-anonymity guarantees that even if an attacker knows the QID values of a record owner, that record owner remains anonymous within a group, known as an *equivalence class*. The parameter k acts as a control for the level of anonymity offered, with larger values enhancing anonymity but potentially reducing data utility. Table 1 provides an example of k-anonymity with $k = 3$.

The main limitation of k-anonymity arises from the potential of similar or identical SA values in the equivalence classes, which can constrain the anonymity protection it offers. Therefore, k-anonymity cannot protect against attribute disclosure attacks.

2.3 Generalization

Generalization stands out as the main non-perturbative technique employed with k-anonymity. Non-perturbative techniques, in general, reduce data information by minimizing or suppressing detail without altering the content, preserving data truthfulness. Generalization enhances data anonymity by replacing QID values with more generalized yet semantically consistent values [37]. For categorical attributes, such as gender or job, specific values can be replaced with more general values using a value generalization hierarchy (VGH) [32]. For each attribute, a VGH is described as a tree structure whose leaves contain the values of the attribute and non-leaf nodes define generalized values. Figure 1 shows a VGH for the attribute *Job*, as an example. In this figure, jobs like surgeon and nurse are generalized to the broader category of medical professional. For numerical attributes, such as age, exact values can be replaced by intervals containing the exact values. For example, the age 45 could be generalized to the interval "[41-60]".

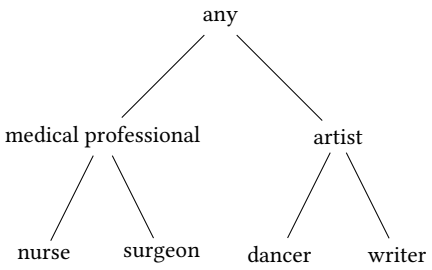


Figure 1: Example VGH for the categorical attribute *job*

While there are various approaches to achieving k-anonymity, our focus in this paper is on k-anonymization based on generalization. This choice is driven by the non-perturbative nature of generalization, which aids in preserving data truthfulness compared to perturbative techniques that often introduce new information. Although suppression (deleting specific QID values and replacing them with a special symbol, e.g., *) is non-perturbative, we exclude it due to its tendency to lower data utility [20]. Therefore, to maintain high data utility, generalization appears to be a more favorable option than suppression. It is important to note that over-generalization can negatively impact data analysis results [26].

2.4 Sampling

Sampling involves retaining only a portion of records from an original dataset and can occur either before (pre-sampling) or after (post-sampling) k-anonymizing the dataset [39]. Pre-sampling reduces the input dataset size for k-anonymization, thereby lowering computing power requirements. Post-sampling allows for advanced techniques that leverage the k-anonymous dataset generated after generalization. Combining both pre-sampling and post-sampling is feasible for very large datasets.

Sampling can be employed using different methods. The most common methods are simple random sampling and stratified sampling [28]. Simple random sampling involves randomly removing the desired number of records, ensuring no bias by definition. However, it does not guarantee equal removal from each equivalence class, leading to an unfair distribution in protection against de-anonymization attacks [39]. Stratified sampling addresses this issue by proportionally removing records from each equivalence class based on its size. This ensures a certain level of uncertainty for each record. As this method relies on equivalence classes, it is only applicable as a post-sampling method.

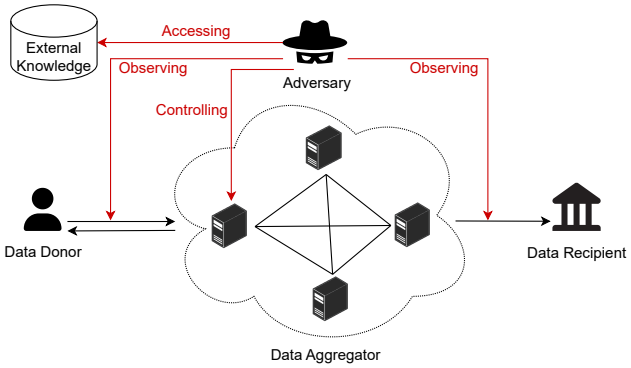


Figure 2: An overview of the system and threat model.

In our protocol, we employ post-stratified sampling, a choice that aligns well with the assumptions we outline for our system model and threat model, as detailed in the next section.

3 MODELS & PROPERTIES

This section describes the system model and the design assumptions of our proposed protocol, *Anonify*. It also discusses the adversary's goal and capabilities and presents the security properties of our protocol.

3.1 System Model

We consider a model that consists of three main components: the data donors (users), the data aggregator, and the data recipients (researchers), see Figure 2.

Data Donors. We assume a set of users U that represent the data donors who participate by sending their data to the data aggregator. Each user $u_i \in U$ has a record r_i that consists of a set of QIDs (personal information) and a set of SAs (medical data). Each QID is denoted as qid_j, and each SA is denoted as $sens_j$. That means r_i can be represented as $r_i = \{qid_1, qid_2, \ldots, qid_x, sens_1, sens_2, \ldots, sens_y\}$.

To guarantee truthfulness at the record level, we assume that every user u_i is truthful and does not falsify their data. Each record r_i collected by the system matches with an existing individual in real life [15].[2]

Data Aggregator. The data aggregator $\mathcal{G}$ is responsible for collecting users' data and making it accessible to the data recipient(s). The responsibilities of the aggregator $\mathcal{G}$ are distributed across n servers, implying that $\mathcal{G}$ is managed by multiple servers to avoid dependence on a single entity. Each of these servers maintains a copy of a database D, and collectively, the servers manage the contents of this database. The servers are assumed to employ methods that enable anonymous communication. Furthermore, the aggregator $\mathcal{G}$ is expected to ensure data anonymity for users through the implementation of k-anonymity and stratified sampling in decentralized settings.

[2]Data truthfulness is crucial in the context of medical data donation because the donated data can be used for treatments or the analysis of the effects of medicines [8]. Falsified data can lead to incorrect treatments and analyses.

Data Recipient. The entity that needs the donated data is referred to as the data recipient $\mathcal{T}$. In some scenarios, there might be several data recipients, such as multiple research institutes. We assume that the recipient gets an anonymized dataset of the users' records (R'). Each record in R' corresponds to a unique user and contains an anonymized information about the user's personal information and medical data. It is important to note that we only consider the case where the exchanged data between users and researchers is unidirectional. In other words, we focus on the situation where researchers collect users' data but do not provide anything in return, such as feedback or analysis results, to the users.

3.2 Threat Model

We consider the presence of an adversary, denoted as $\mathcal{A}$, whose goal is to de-anonymize users by identifying their SA values to gain insights into their health details. $\mathcal{A}$ can have background knowledge about some of the users, specifically information about their QID values. The set U' represents users about whom $\mathcal{A}$ has no knowledge of their QID values. Additionally, since each user is assumed to generate their SA values locally and these values are only known to the user, $\mathcal{A}$ is assumed to have no information about the SA values of any users in the set U.

$\mathcal{A}$ is a global passive adversary who is capable of observing all incoming and outgoing network traffic in the system. However, $\mathcal{A}$ is not able to manipulate the traffic, such as dropping, altering transmitted messages, or injecting new messages. Neither can it gain any information about the actual content of messages while transmission due to message end-to-end encryption. Further, it lacks the capability to associate users based on message size, given our assumption of fixed message size.

We assume that the adversary collaborates with the data recipients; therefore, we consider the data recipient(s) as untrusted. Besides, $\mathcal{A}$ can have control over a subset of $\mathcal{G}$'s servers, meaning at least one of the servers must be honest. Even under the assumption that a subset of $\mathcal{G}$'s servers may be malicious, $\mathcal{G}$ is expected to maintain honesty in executing the protocol. Without this assumption, anonymity is unaffected (see Section 5), but availability could be compromised as malicious servers may deny the service by manipulating their database instances. All users are also assumed to be honest, as this is important for ensuring data truthfulness and maintaining system availability. It is worth noting that a malicious user cannot manipulate others' data but can submit false data or compromise availability by submitting corrupted DPF shares.

3.3 Security Properties

Anonify aims to provide the following properties:

3.3.1 Anonymous Communication. Our protocol is designed to protect communication between users (data donors) and $\mathcal{G}$ by providing the following two anonymity properties: *sender anonymity* and *receiver anonymity*. These properties prevent $\mathcal{A}$ from de-anonymizing users based on communication metadata such as IP addresses, or the time of sending or receiving.

Informally, sender anonymity is the property where, $\mathcal{A}$ cannot determine which user in U' wrote specific messages in the database D any better than making random guesses.

Definition 2 (sender anonymity). *When $\mathcal{A}$ lacks prior knowledge about the message's content, the protocol guarantees sender anonymity. For each message within D, sender anonymity is achieved if the protocol ensures that $\mathcal{A}$ cannot significantly reduce the probability of accurately identifying the user $u_i \in U'$ responsible for writing a specific message in D to less than $1/|U'|$.*

Essentially, this implies that the adversary cannot de-anonymize u_i (i.e., associate a message with a specific user) more effectively than random guessing from the set of users U'. The strength of the sender anonymity property depends on the size of U', where a larger $|U'|$ implies a larger anonymity set size, thereby ensuring stronger anonymity. To ensure a minimum level of anonymity, the size of U' should be larger than or equal to a certain threshold.

During the protocol execution, particularly in the registration phase, each user will need to retrieve specific information from $\mathcal{G}$ to proceed to the publishing phase. This information should be obtained anonymously. As a result, the protocol provides the receiver anonymity property for users, ensuring that no adversary can learn which piece of information a user is interested in retrieving from $\mathcal{G}$.

Definition 3 (receiver anonymity). *When $\mathcal{A}$ lacks prior knowledge regarding the content that the user u_i wants to retrieve from $\mathcal{G}$, the protocol provides receiver anonymity. This is achieved when the protocol ensures that $\mathcal{A}$ has only a negligible probability of successfully deducing the specific data $u_i \in U'$ intends to retrieve, with this probability being close to $1/|U'|$.*

Our protocol guarantees that all users can access the data of interest from the aggregator while minimizing the likelihood of $\mathcal{A}$ successfully discovering which data u_i is retrieving down to $1/|U'|$.

3.3.2 Data Anonymity. Our protocol protects users at the data level by providing the following two anonymity properties: *k-anonymity* and *sensitive attribute (SA) uncertainty*. These properties protect users from data de-anonymization attempts, particularly by defending against identity and attribute disclosure attacks that may target the donated data.

k-anonymity necessitates that a minimum of k individuals have identical QID values in the anonymized dataset R'. In this manner, even if $\mathcal{A}$ acquires knowledge of the QID values associated with a particular record owner, that individual still retains anonymity concerning QID values within their equivalence class.

Definition 4 (k-anonymity). *The protocol provides k-anonymity if for each unique combination of QID values that is present in R', there exist at least k-1 other records in R' with the identical combination of these QID values.*

The k-anonymity property serves as a safeguard against identity disclosure attacks. The k parameter represents the minimum size of an equivalence class within the anonymized dataset R'. The value of k determines the level of anonymity offered, where higher values of k lead to stronger anonymity protection.

Relying solely on the k-anonymity property may not provide sufficient data protection for users. While k-anonymity can guarantee protection against identity disclosure attacks, it cannot ensure protection against attribute disclosure attacks. These can compromise the data anonymity of users when there is a lack of diversity in SA values within equivalence classes in R'. In a scenario where $\mathcal{A}$ knows the QIDs for a user u_i, it can identify the equivalence class to which u_i belongs. If all records within this class share identical SA values, $\mathcal{A}$ can then learn the SA values of u_i. To address this, our protocol introduces an additional property known as SA uncertainty. This property ensures that $\mathcal{A}$ cannot determine whether u_i's record is among the records in R'. Consequently, even if all records in R' share a specified SA value, $\mathcal{A}$ cannot ascertain if u_i has this value. This property is achieved in our protocol by applying stratified sampling.

Definition 5 (sa uncertainty). *The protocol provides SA uncertainty if the probability that $\mathcal{A}$ can guess that u_i's record is one of the records in R' is ρ. The value of ρ is calculated as the ratio of the number of records in u_i's equivalence class before sampling to the number after sampling.*

The parameter ρ defines the protection level granted by the SA uncertainty property. Smaller ρ values indicate greater uncertainty for $\mathcal{A}$ regarding the SA values of users, resulting in higher protection.

SA uncertainty, like t-closeness and ℓ-diversity, protects against attribute disclosure attacks. However, both t-closeness and ℓ-diversity require centralization, whereas SA uncertainty can be achieved by a decentralized system. Additionally, SA uncertainty ensures that adversaries cannot determine the presence of a user's SA values in R', whereas t-closeness and ℓ-diversity do not conceal the presence of a user record in R'.

4 PROTOCOL ARCHITECTURE

In this section, we describe *Anonify*, our decentralized protocol designed to enable medical data donation. *Anonify* operates in two phases: the registration phase and the publishing phase. The main steps of the protocol are illustrated in Figure 3.

4.1 Registration Phase

The protocol begins with a registration phase, during which users are required to submit their personal information (QID values) to $\mathcal{G}$. To prevent $\mathcal{G}$ from linking users to their submitted data through the exploitation of communication metadata, our protocol employs the DPF method. Other anonymous communication systems, such as onion routing or mix networks, can also be used to maintain unlinkability between users and the data they send to the aggregator. However, these alternatives may have weaker security guarantees compared to the DPF method we use, as they are known to be vulnerable to traffic analysis attacks by an adversary controlling a substantial portion of the network or monitoring multiple links within it [9].

Algorithm 1 depicts the details of the registration phase. The users first write their data anonymously in D using DPF. After a specific number of users contribute their records into D, the $\mathcal{G}$'s servers collaboratively share and combine their database instances to unveil the content in D. Subsequently, they apply k-anonymity, categorizing records into equivalence classes based on the similarity of QID values.[3] This categorization guarantees that each class $EC_j \in$

[3]All servers are assumed to execute the protocol honestly, allowing any individual server to perform k-anonymization. However, advocating for uniform execution by all servers ensures process integrity and consistent results.

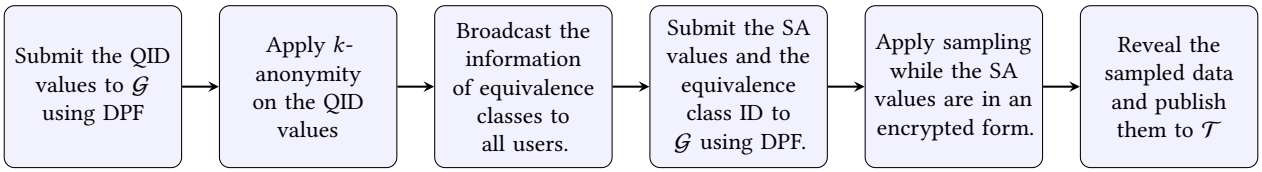


Figure 3: The main steps of the *Anonify* protocol, with the initial three steps corresponding to the registration phase and the subsequent three steps associating with the publishing phase.

EC consists of a minimum of k records. The servers allocate an identifier e_j to each class EC_j, followed by broadcasting a message to all users. [4] This message contains a list of class identifiers and the personal identifiers of users assigned to each class. Each user independently determines their class identifier by checking which equivalence class their personal identifier p_i is associated with. It is important to note that each user generates their personal identifiers locally.

Algorithm 1 Registration Phase

1. **Prepare Registration Message.** For each user u_i, let p_i be a randomly generated identifier, and $qr_i = \{qid_1, qid_2, \ldots, qid_x\}$ represents a record of QID values. The user u_i creates a message $m_i = (p_i, qr_i)$.
2. **Submit DPF Shares.** Each user u_i:
   - generates DPF shares: $\{f_1, \ldots, f_n\} = \mathsf{GenDPF}(m_i, t)$, where t is a random index in D.
   - submits one share to each of the n servers of $\mathcal{G}$. The share is encrypted using the receiving server's public key.
3. **Reveal the Database Content.** Each server of the n servers of $\mathcal{G}$:
   - uncompresses the u_i's DPF share and adds the result to its instance of D.
   - If the number of new users registering on the servers exceeds a predefined threshold, U is defined as this group of new users.
   - exchanges its database instances with the other servers and combines all instances to get the messages written to D by users in U.
4. **Apply k-anonymity.** The servers:
   - group the users' records (i.e., qr_i) into equivalence classes based on the similarity of QID values.
   - apply generalization to the records in each equivalence class $EC_j \in EC$.
   - allocate a distinct identifier e_j to each equivalence class EC_j.
5. **Broadcast Classes IDs.** The servers send, to all users, a message that contains each class's identifier e_j along with the set of users' identifiers attached to the records in EC_j.
6. **Get Class ID.** Each user u_i looks for her identifier p_i in the broadcasted message and finds the corresponding equivalence class identifier (e_j).

[4] Any of the servers can perform this step. Another approach involves dividing the set of users among the servers, with each server broadcasting the message to only a specific subset of users.

In Figure 4, Table (a) presents an example of the data that the servers have after revealing the messages users wrote in D, while Table (b) illustrates the equivalence classes generated through the k-anonymization step. Each equivalence class is associated with a class identifier e_j, QID values representing users within the class, and the identifiers of users belonging to that specific class. As shown in the table, users belonging to each class share the same generalized personal information (QID values) with all other users in the same class, making them indistinguishable within this specific equivalence class in terms of QID values.

Collisions. It is crucial to consider the problem of collision when users generate their random personal identifiers. Since each user u_i generates p_i locally, this can potentially create a situation where two users end up with the same identifiers. To significantly reduce the likelihood of this happening, we recommend that each user generates a random 128-bit number as their identifier.

Another collision issue can arise when users write their messages anonymously using DPF. As each user independently and randomly chooses the index to write their message in the database D, there is a risk of concurrent selections leading to two users attempting to write messages to the same index. In such cases, the content of these messages becomes irretrievable as one user's message overwrites another's. To address this issue, the size of D should be configured in a manner that significantly minimizes the probability of collisions. In other words, the size of D should be sufficient to accommodate the anticipated number of messages while maintaining a high probability of writing success without encountering collisions. In the work presented in [7], a formula is provided to compute the expected writing success rate for a given database size ℓ. This formula can assist in choosing a size that minimizes the likelihood of collisions:

$$SuccessRate \approx 1 - \frac{ReqCount}{DBSize} + \frac{1}{2}\left(\frac{ReqCount}{DBSize}\right)^2$$

For example, to handle writing requests from 100,000 users ($ReqCount = 100,000$ with each user having one writing request) and achieve an expected success rate of 90%, the database size $DBSize$ should be set to 1,000,000.

4.2 Publishing Phase

After users complete the registration phase, they proceed to the publishing phase, where they send their SA values (i.e., medical data) to $\mathcal{G}$. In this phase, communication between users and $\mathcal{G}$ is also established through the DPF method. Further, *Anonify* safeguards against the inference of users' SA values during this phase by employing protection against attribute disclosure attacks. This

PID	Sex	Age
3	F	32
5	F	26
8	M	45
9	M	37
12	M	42
16	F	28

(a)

CID	Sex	Age	PIDs
1	F	25-35	3, 5, 16
2	M	35-45	8, 9, 12

(b)

Figure 4: The information that $\mathcal{G}$'s servers have during the registration phase. PID stands for the user's random identifier (p_i), while CID stands for the class identifier (e_j).

CID	Test result
1	Positive
1	Negative
1	Positive
2	Negative
2	Negative
2	Negative

(a)

CID	Test result
1	Negative
1	Positive
2	Negative
2	Negative

(b)

Figure 5: The information that $\mathcal{G}$'s servers have during the publishing phase includes Table (a) in encrypted form and Table (b) in plain text.

protection is achieved through the use of stratified sampling, where servers only reveal a portion of records within each equivalence class. Leveraging DPF, *Anonify* applies stratified sampling in a decentralized manner and on encrypted records, eliminating the necessity to trust the aggregator with entire records before sampling.

Algorithm 2 describes the steps for the publishing phase. First, each user selects a random index t' in the database D and anonymously writes their class identifier at this index. Next, servers randomly select a fixed number of indices in D associated with each class. They designate these indices as locations where records will not be revealed (note that this occurs before users submit their actual records containing their SA values). Subsequently, each user anonymously writes their records to D at the same index t'. The servers then delete the records at indices in D marked not to be revealed; this process occurs while the records are still in an encrypted form. Therefore, servers remain unaware of the content of unreleased (deleted) records. After the sampling process, servers create a dataset from the remaining records in D and transmit this dataset to $\mathcal{T}$.

An example of the sampling is illustrated in Figure 5, where Table (b) represents a sample of the data in Table (a). In this example, only two records are released for each class, indicating that one record is deleted in each class.

Multiple Iterations of Medical Data Donation. In certain scenarios, data recipients (researchers) may require users to periodically submit their medical data (SA values). For instance, consider

Algorithm 2 Publishing Phase

1. **Reserve Index.** Each user $u_i \in U$:
   - generates DPF shares: $\{f_1, \ldots, f_n\} = \mathsf{GenDPF}(e_j, t')$, where t' is a random index in D.
   - submits one share to each of the n servers of $\mathcal{G}$. The share is encrypted using the receiving server's public key.
2. **Identify Indices Related to Each Class.** The servers:
   - add the received shares to their database instances and collaboratively reveal the content of D after all users in U submit their shares.
   - identify indices that contain a similar class identifier e_j.
3. **Choose the Unconsidered Indices.** The servers randomly select μ indices associated with each e_j. The set of all selected indices is defined as Z.
4. **Send SA Values.** Each user $u_i \in U$:
   - generates DPF shares: $\{f_1, \ldots, f_n\} = \mathsf{GenDPF}(sr_i, t')$, where $sr_i = \{sens_1, sens_2, \ldots, sens_y\}$ represents the u_i's record of SA values.
   - submits one share to each of the n servers of $\mathcal{G}$. The share is encrypted using the receiving server's public key.
5. **Apply Stratified Sampling.** Each server:
   - waits until adding to its database instance the shares received from every $u_i \in U$.
   - deletes the content of the indices in its database instance that are part of Z (refer to step 3).
   - exchanges its updated instance with the other servers and combines all instances to unveil the remaining records in D.
6. **Create the Anonymized Dataset.** The servers create the dataset R' by grouping together the records in D written in indices associated with the same class identifier e_j.
7. **Forward to the Recipient.** The servers transmit R' to the data recipient $\mathcal{T}$ along with the corresponding QID values that represent each class.

the "safevac" app introduced by the Paul-Ehrlich Institute (PEI) during the COVID-19 pandemic for a study on the vaccine's effects [25]. This study involves users downloading the app and responding to surveys at specific intervals following vaccination. In such cases, the typical single execution of the publishing phase is replaced by multiple iterations. This necessitates certain adjustments to the publishing phase. One key requirement is enabling data recipients to link data points from the same source, as this is pivotal for effective data analysis. To achieve this while safeguarding user anonymity, the following updates should be made to the publishing phase:

In the initial iteration where users submit their SA values, the steps of the publishing phase as outlined in Algorithm 2 must be followed. However, a crucial modification is required in step 4. Specifically, the message passed to the GenDPF function should be $m_i = (p'_i, sr_i)$ instead of passing sr_i only, where p'_i denotes a locally generated random identifier by a user u_i. Importantly, p'_i should be distinct from the identifier p_i created by u_i during the registration phase. This differentiation is important, as any similarity between these identifiers would enable servers to link the QID values submitted by the user in the registration phase with

the SA values provided during the publishing phase. Such a linkage would compromise user anonymity, undermining the protocol's fundamental objective.

In the following iterations where users need to provide new SA values to the data recipients, each user must consistently submit their message to the same index denoted as t', which they initially selected during the first iteration of the publishing phase. Additionally, in these subsequent iterations, only steps 4 through 7 of Algorithm 2 should be carried out. This implies that, across all iterations, the protocol always removes the records submitted to the indices that are part of the set Z, and Z is defined only once in the first iteration of the publishing phase. As a result, the protocol releases records from the same set of users in every iteration. This approach effectively prevents any information leakage to $\mathcal{A}$ across iterations regarding which users have records that have been published and which users have had their records removed by the protocol, thus protecting against intersection attacks [16, 17].

5 SECURITY ANALYSIS

In this section, we show that *Anonify* reaches the security properties that are defined in Section 3.3. *Anonify* achieves sender anonymity and receiver anonymity only for the set of users $U' \in U$ (see Section 3.2). It achieves k-anonymity and SA uncertainty for the set of all users U.

Sender Anonymity. *Anonify* ensures sender anonymity for users in U' during both the registration and publishing phases. To achieve sender anonymity, the users should be unlinked to the messages they write in D. In our protocol, this is achieved through secret sharing based on DPF. The anonymity guarantees of DPF were proven in [7].

We prove the sender anonymity property in our protocol by showing that $\mathcal{A}$ cannot use any of its abilities to compromise this property.

- *Passive Observation.* In each protocol phase, all users adhere to the protocol, ensuring that they each send the same number of shares to the servers. All messages exchanged between users and servers have the same size and are encrypted using the public key of the receiving server. The servers add all incoming shares to the database D and disclose all messages written by all users in D at once. Therefore, $\mathcal{A}$ is unable to link messages to senders through passive observation of requests between users and servers.
- *Server Corruption.* As per our assumptions, $\mathcal{A}$ has the capability to corrupt all but one of the $\mathcal{G}$'s servers. $\mathcal{A}$ can link a user with the share they send. However, $\mathcal{A}$ can learn at most $n-1$ out of the n DPF-shares from users. Existing literature provides formal proof that combining $n-1$ shares does not disclose any information about the content of the enclosed message [19].

Receiver Anonymity. *Anonify* guarantees receiver anonymity for users in U'. This property is only required during the registration phase, as users need to retrieve their corresponding equivalence class identifier. Receiver anonymity is achieved through a broadcast mechanism that ensures $\mathcal{A}$ cannot link users to their class identifiers.

We prove the receiver anonymity property in our protocol by demonstrating that $\mathcal{A}$ is incapable of utilizing any of its abilities to break this property.

- *Passive Observation.* Messages are only delivered to users when the servers broadcast a message containing the equivalence classes identifiers and the list of the personal identifiers of users assigned to each equivalence class. Since all users receive the same message from the servers, $\mathcal{A}$ cannot determine the class identifier of a user $u_i \in U'$ by simply observing communication between users and servers.
- *Server Corruption.* Since users generate their personal identifiers locally, $\mathcal{A}$ is unable to associate $u_i \in U'$ with their class identifier in the message by leveraging u_i's personal identifier. Furthermore, since $\mathcal{A}$ lacks prior knowledge about the QID values of u_i, it cannot establish a link between $u_i \in U'$ and their equivalence class identifier in the message by exploiting u_i's QID values.

k-anonymity. *Anonify* ensures that the dataset R' achieves k-anonymity. With this property, $\mathcal{A}$ cannot link a user $u_i \in U$ to their SA values in the dataset R' by exploiting the u_i's QID values, subject to the restriction that the size of the equivalence class of u_i is at least k. [5]

We prove the k-anonymity property in our protocol by showing that $\mathcal{A}$ cannot use any of its abilities to break this property.

- *Background Knowledge about QID values.* The dataset R' consists of equivalence classes, each containing a minimum of k records, with each record within an equivalence class containing SA values. Due to the shared QID values among all records within the same class, $\mathcal{A}$ is unable to identify a specific record for a user u_i, even if $\mathcal{A}$ has knowledge of the QID values of u_i and can specify the class to which u_i belongs.
- *Server Corruption.* Although $\mathcal{A}$ may have control over specific servers, it is unable to manipulate or interfere with the protocol execution, as even malicious servers are obligated to honestly execute the protocol. As a result, the dataset R' is constructed in an honest manner, with each record sharing identical QID values with at least k other records.

SA Uncertainty. *Anonify* ensures SA uncertainty, leaving $\mathcal{A}$ uncertain about the inclusion of SA values for user $u_i \in U$ in R'. The capability of $\mathcal{A}$ is limited to probabilistically guessing whether the SA values of user u_i are present in R', with a probability denoted by ρ. We prove the SA uncertainty property in our protocol by demonstrating that $\mathcal{A}$ is unable to leverage any of its capabilities to break this property.

- *Background Knowledge about QID values.* $\mathcal{A}$, with background knowledge of u_i's QID values, can identify the equivalence class in R' to which u_i belongs. However, even with knowledge of all classes and their associated users, $\mathcal{A}$ cannot be certain about the inclusion of u_i's specific record within the records released in an equivalence class in R'. This uncertainty arises due to the protocol's employment of stratified sampling. The protocol selects random records from each

[5] Due to sampling, the sizes of equivalence classes in R' are smaller than their sizes in the registration phase. Therefore, this should be considered when setting the value of k during registration to ensure that the final k value in R' is not very low.

equivalence class to be released in R'. Thus, the presence of the u_i's record among the released records becomes probabilistic, with the likelihood determined by the ratio ρ (see Section 3.3).

- *Server Corruption.* All servers, including potentially malicious ones, adhere to honest execution of the protocol. The use of DPF and the local selection of the message index in D by each user u_i prevents $\mathcal{A}$ from identifying the specific index to which u_i wrote their SA values in D. The stratified sampling step is done while the records are in an encrypted form in D which means the $\mathcal{G}$'s servers cannot determine which records that are deleted. Single servers cannot decrypt records before the stratified sampling step because the DPF method ensures that only collaborative decryption is possible. The stratified sampling protects against the $\mathcal{A}$'s ability to determine whether or not u_i's record containing their SA values is among the records in R'. Therefore, even if all the records in R' share a specified value for SA, $\mathcal{A}$ cannot be sure if u_i has this value because u_i's records might be one of the deleted records.

Impact of Weakening Assumptions on Anonymity. We assume that all servers, including potentially malicious ones, faithfully adhere to the protocol during execution (see Section 3.2). This assumption is made to ensure system availability, not anonymity, as the anonymity protection remains uncompromised without this assumption. The reason behind this is that the protocol relies on the following:

- DPF: This method guarantees that the content of submitted messages can only be revealed when all servers collaborate, with the condition that at least one server is honest. In the scenario where n-1 servers cooperate, the disclosure of message content becomes impossible. Therefore, sender anonymity and SA uncertainty cannot be broken, even in the presence of malicious servers deviating from the protocol.
- Equivalence classes agreement: The protocol requires that all servers reach the same set of equivalence classes. This ensures that malicious servers cannot manipulate the k-anonymization step or the list containing each class's identifier and the associated personal identifiers without detection by honest servers. Thus, malicious servers are unable to compromise k-anonymity and receiver anonymity.

To demonstrate the deterministic guarantees of *Anonify*, we assume user honesty and $\mathcal{A}$'s lack of knowledge regarding SA values. A malicious user cannot compromise the anonymity of other users unless they collude with $\mathcal{A}$ and disclose their SA values to it. However, if the malicious users refrain from collusion with $\mathcal{A}$, their influence is limited to affecting system availability and data truthfulness, not anonymity.

When $\mathcal{A}$ lacks knowledge of SA values, the certainty of $\mathcal{A}$ regarding u_i's record being in R' is calculated deterministically. It is expressed as the ratio of the number of records in u_i's equivalence class before sampling to the number after sampling. When $\mathcal{A}$ is aware of the SA values of some users in the equivalence class to which u_i belongs, the certainty about a u_i's record being in R' becomes entirely probabilistic and complex. It depends on factors such as the number of users in the class whose SA values are unknown to $\mathcal{A}$, and whether the records of these users are part of the sampled dataset R'. As sampling is done randomly within each class, the records in R' may be exclusively drawn from users with unknown SA values, from users with known SA values, or from a combination of both sets of users. Adding to the complexity, the certainty of $\mathcal{A}$ is also affected by the potential scenario where users with unknown SA values may share similar values with users whose values are known. This can significantly complicate the differentiation between records in R' belonging to users with known SA values and those belonging to users with unknown SA values.

6 EVALUATION

Anonymization approaches that alter data to meet anonymity needs often do so at the expense of data utility. Therefore, to evaluate such approaches, it's vital to assess both their anonymity impact and the utility of the anonymized data. In this section, we provide the performance results of *Anonify* in terms of anonymity and data utility.

As mentioned earlier, our protocol aims to ensure both anonymous communication and data anonymity. In our evaluation, we focus on assessing the data anonymity aspect. On the one hand, the protocol's guarantees regarding anonymous communication are deterministic, and their verifiability is demonstrated in Section 5. On the other hand, the bandwidth overhead in our protocol generally remains low, given that the number of messages users need to submit or receive is limited (see Section 4). Moreover, the use of DPF does not introduce high bandwidth overhead, as each share a user needs to send should have a bitlength of $O(\lambda \cdot \log \ell + |m|)$[13]. For example, the share size is expected to be 10.096 KB when the security parameter λ is set to 128 (as in[13]), the message size $|m|$ is 10 KB, and the database size ℓ is 1,000,000. Additionally, concerning latency overhead, it is crucial to note that the medical data donation scenario typically tolerates higher latency. Furthermore, employing DPF for anonymous writing has been proven to introduce low latency, even with a large user base, as demonstrated in [13, 18].

Nevertheless, it is important to emphasize that multiple parameters influence the latency and bandwidth overhead resulting from employing *Anonify*. A key parameter is n, representing the number of servers running $\mathcal{G}$. With an increase in the number of servers, more shares must be sent, and more database instances need to be combined, resulting in higher bandwidth overhead and latency. Another significant factor is the size of U, as it directly impacts the size of the database D. A larger U leads to a substantially larger database, causing increased latency when adding shares to a database instance and necessitating more computation time to combine all instances.

6.1 Dataset

In our evaluation, we simulate the records that the users send to $\mathcal{G}$ using the diabetes prediction dataset [6]. This dataset contains 100,000 records, each representing a distinct individual. It includes medical and demographic data, along with the diabetes status (i.e., whether individuals have diabetes or not). The dataset consists of features such as age, gender, body mass index (BMI), hypertension, heart disease, smoking history, HbA1c level, and blood glucose

[6] https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset

level. It can be used to train machine learning models to predict diabetes in patients based on their medical history and demographic information. Further, researchers can use this dataset to investigate the links between various medical and demographic factors and the chance of developing diabetes.

In our evaluation, we use the entire dataset, i.e., the size of U is 100,000 individuals. We designate the attributes—age, gender, and BMI—as QIDs because we assume that they can potentially be acquired by an adversary, for example, through information available on social networks. Conversely, we classify the remaining attributes as SAs since they pertain to vital health-related information that we presume the adversary does not have access to.

6.2 Data Utility Assessment

We assess the effectiveness of our protocol by measuring data utility using the Normalized Certainty Penalty (NCP) metric [43]. Additionally, we evaluate data utility by examining the change in data distribution after anonymization. Furthermore, we assess the performance of machine learning classification algorithms on the anonymized datasets produced by our protocol.

6.2.1 Information Loss. The NCP metric can quantify the information loss resulting from anonymization. This metric captures the uncertainty created by generalization [43]. It is very suitable to be used when the exact usage of the data is not well defined yet by the recipients [22], i.e., general-purpose data.

Let a dataset R with quasi-identifiers (num_1, num_2, ..., num_x, cat_1, cat_2, ..., cat_y), where (num_1, num_2, ..., num_x) are numerical attributes and (cat_1, cat_2, ..., cat_y) are categorical attributes. NCP defines the uncertainty for both of these attribute types. For a record $r_i \in R$, the NCP is computed as follows:

$$NCP(r_i) = \frac{\sum_{j=1}^{x} NCP_{num_j(r_i)} + \sum_{j=1}^{y} NCP_{cat_j(r_i)}}{x + y}$$

Where $NCP_{num_j(r_i)}$ represents the NCP of r_i with respect to num_j, and $NCP_{cat_j(r_i)}$ denotes the NCP of r_i with respect to cat_j.

The NCP for the entire dataset R is the sum of NCP values for all records:

$$NCP(R) = \frac{\sum_{i=1}^{|R|} NCP(r_i)}{|R|}$$

The data utility of the k-anonymized dataset is influenced by both the value of k and the specific k-anonymization method applied. Therefore, in our evaluation using NCP, we conducted tests with various k values. Further, we considered three different k-anonymity methods: Mondrian [23], one-pass k-means (OKA) [24], and ARX [29].

Figure 6 shows the results of our data utility experiments using the NCP metric. As depicted in the figure, we systematically varied the value of k, starting with an initial value of 50 and incrementing it in steps of 50. These experiments encompassed the entire dataset. The results obtained with the ARX method significantly outperformed those of the Mondrian and the OKA, even with higher k values. Moreover, our observations revealed a consistent increase in the NCP values across all methods as the k value increased. This phenomenon can be attributed to the fact that larger k values lead to a larger minimum size of equivalence classes. Thus, it becomes increasingly challenging for records within the same equivalence class to reach an agreement on attribute values. Consequently, more generalization is required to satisfy the k-anonymity criterion, resulting in a reduction in data utility. These findings underscore the inherent trade-off between data utility and anonymity.

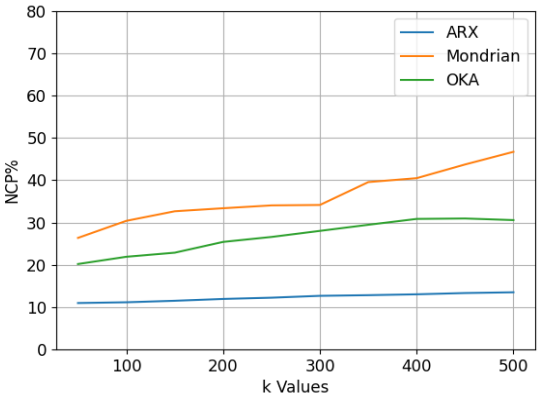


Figure 6: The impact of the k value and the k-anonymization method on data utility.

Since ARX produces the best results, we base our analysis on its results in all the experiments discussed in the following subsections. Furthermore, in all the upcoming experiments, we set the value of k to 250, as this choice strikes a favorable balance between anonymity and data utility.

6.2.2 Data Distribution. We compare the data distribution among three versions of the dataset: the original dataset, a k-anonymized dataset (referred to as "Anonymized" in the figures, representing the original dataset after k-anonymization), and a sampled anonymized dataset (representing R', i.e., displaying only a percentage of records from every equivalence class in the k-anonymized dataset). This data distribution comparison provides an indication of how the data has changed and illustrates the impact of anonymization.

In the experiments, when the sampling percentage is set at 70%, it indicates that μ (see Section 4.2) equals $100 - 70 = 30$, meaning 30% of the records in each equivalence class in the k-anonymized dataset were deleted. Lower sampling percentages imply more protection but typically at the expense of data utility. It's worth highlighting that in all our sampling experiments, we ran each experiment 10 times and computed the average results. This is necessary because the selection of records to be released within each equivalence class is performed randomly.

Figure 7 illustrates the average blood glucose levels for different age ranges across the original dataset, the k-anonymized dataset, and the sampled anonymized dataset (with a sampling percentage of 50%). Notably, the results show that both the k-anonymized dataset and the sampled anonymized dataset closely resemble the original dataset. This suggests a minimal impact of anonymization on the data distribution, as even when only 50% of records in the k-anonymized dataset are sampled, the distribution is still not noticeably affected. A similar finding is demonstrated in Figure 8 which depicts the number of individuals with hypertension based

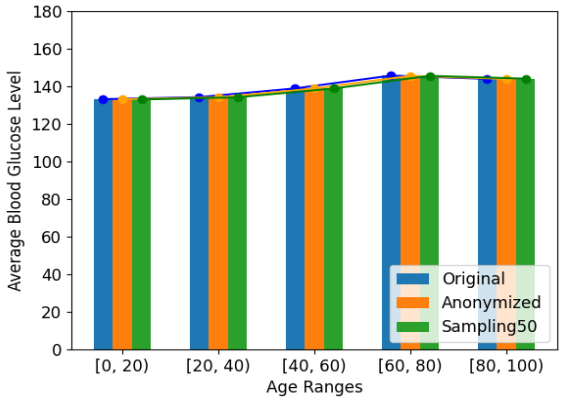


Figure 7: The average blood glucose level within different age ranges (k=250, Sampling percentage=50%).

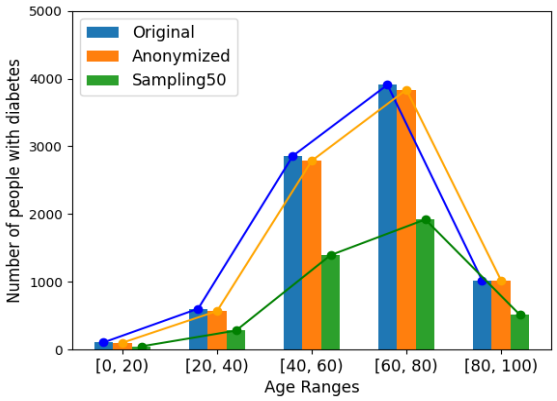


Figure 9: The average number of people with diabetes within different age ranges (k=250, Sampling percentage=50%).

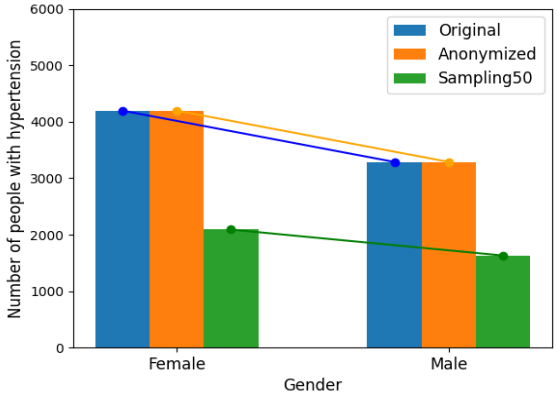


Figure 8: The average number of people with hypertension within different gender groups (k=250, Sampling percentage=50%).

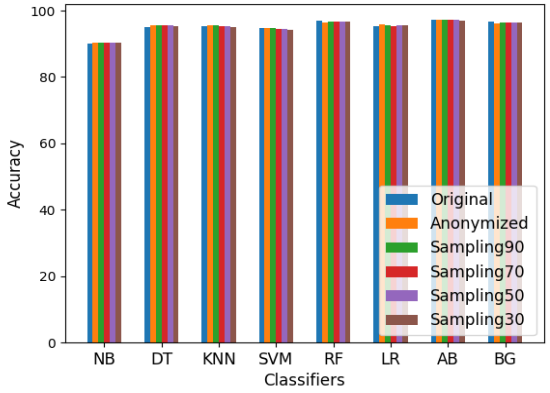


Figure 10: Accuracy (k=250, Sampling percentage=90%, 70%, 50%, and 30%).

on the gender attribute. The sampled anonymized dataset exhibits slight variations but remains well-aligned with the other datasets.

However, the sampled anonymized dataset appears to induce more noticeable alterations in the distribution of individuals with diabetes compared to the original dataset for older age ranges, particularly for age 60 years and above, as shown in Figure 9. This difference in data distribution is influenced by the chosen sampling percentage. Lower percentages, such as 50%, result in more deviations, while higher (e.g., 80%) percentages may yield only slight differences. This highlights the importance of carefully selecting the sampling percentage to achieve anonymity preservation without undermining data quality. In summary, the results demonstrate the effectiveness of our protocol, which integrates k-anonymity and stratified sampling, in maintaining anonymity without jeopardizing data distribution, especially when the parameters are carefully chosen.

6.2.3 Machine Learning Classifiers' Performance. Data recipients may require training machine learning models on the dataset generated by *Anonify* for diabetes prediction. Thus, we assess the performance of machine learning classification algorithms on the anonymized datasets produced by *Anonify*. More specifically, we compare the performance of eight well-known machine learning classification algorithms on the original dataset, the k-anonymized dataset, and different sampled anonymized datasets (with sampling percentages: 90%, 70%, 50%, and 30%). The considered algorithms include Decision Tree (DT), Naïve Bayes (NB), k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Random Forest (RF), Logistic Regression (LR), AdaBoost (AB), and Bagging (BG). These algorithms offer various trade-offs in terms of simplicity, accuracy, robustness, and sensitivity to noise. For a detailed understanding of each, refer to [1, 3].

We assess the classifiers' performance in terms of Accuracy, Precision, Recall, and F1-score, which are defined based on True

Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) values. These four metrics are computed as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \quad , \quad \text{Precision} = \frac{TP}{TP + FP} \quad ,$$
$$\text{Recall} = \frac{TP}{TP + FN} \quad , \quad \text{F1-Score} = 2 \cdot \frac{\text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}$$

In our experiment, we designated the "diabetes" attribute in the dataset as the class label, with values "1" indicating diabetes and "0" representing the absence of diabetes. We partitioned each dataset version into a 75% training set and a 25% test set. Subsequently, we trained the classifiers using the training data and evaluated their performance on the test dataset.

Figure 10 shows the accuracy scores, serving as indicators of the classifiers' proficiency in data classification, with higher values denoting superior performance. The results for the original dataset, k-anonymized dataset, and different sampled datasets show no noticeable differences, as they all exhibit close accuracy scores. Even when sampling only 30% of records from each equivalence in the k-anonymized dataset, the accuracy score remains high and is comparable to or slightly better than the result of the original data. This demonstrates the protocol's capability to be employed without negatively impacting classification accuracy.

Furthermore, this trend extends to recall, precision, and F1 score (as shown in Figure 13, 14 and 15 in Appendix A), reinforcing the fact that applying k-anonymity and sampling maintains the good performance of these metrics, as the results for sampled anonymized datasets remaining closely aligned with or better than those of the original dataset.

6.3 Data Anonymity Assessment

Data anonymization techniques (e.g., k-anonymity and differential privacy) involve altering or removing information in datasets to prevent the identification of individuals [27]. While effective in reducing direct identification risks, they cannot guarantee complete immunity from re-identification risks [14]. To assess re-identification risks in datasets anonymized by Anonify, we employed the Journalist Risk and Certainty metrics, which are commonly considered for this type of assessment [29, 39].

6.3.1 Journalist Risk (JR). It measures the probability of linking a specific record r_i to a targeted user [12]. Let the equivalence class $EC_j \in EC$ is denoted as $EC_j^{\mu} \in EC^{\mu}$ after the sampling step, where $EC^{\mu} \subset EC$. JR is calculated by multiplying the probability that r_i remains in the equivalence class EC_j after sampling (i.e., $\frac{|EC_j^{\mu}|}{|EC_j|}$) with the probability that the attacker selects the correct record from that sampled equivalence class (i.e., $\frac{1}{|EC_j^{\mu}|}$) :

$$JR(r_i) = \frac{|EC_j^{\mu}|}{|EC_j|} \cdot \frac{1}{|EC_j^{\mu}|} = \frac{1}{|EC_j|}, \text{ where } r_i \in EC_j^{\mu}$$

The journalist risk serves as an indicator of the level of risk associated with the anonymization process. A lower index implies more protection, and vice versa. Figure 11 provides an evaluation of the journalist risk index on the k-anonymized dataset and different sampled anonymized datasets. The journalist index results for all

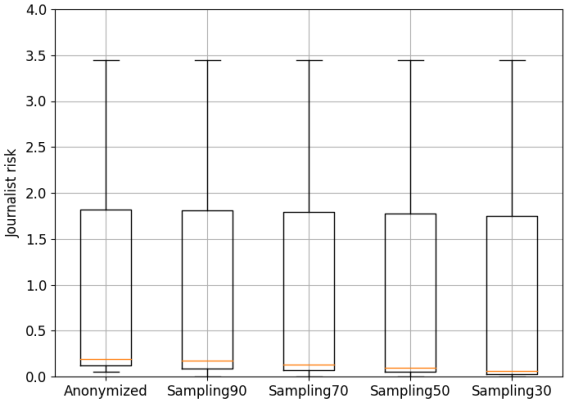


Figure 11: Journalist risk (k=250, Sampling percentage=90%, 70%, 50%, and 30%).

versions of the datasets exhibit very low values, indicating a higher level of anonymity. Additionally, as expected, reducing the number of records released, as seen in lower sampling percentages, leads to lower journalist index values. For example, at a 30% sampling percentage, the average journalist index reaches a significantly low value of 0.057075, indicating a substantially reduced risk.

6.3.2 Certainty Metric. This metric represents the likelihood that a record r_i is contained within the anonymized dataset [39]. It is computed as follows:

$$C(r_i) = \frac{|EC_j^{\mu}|}{|EC_j|}, \text{ where } r_i \in EC_j^{\mu}$$

For each record r_i in the original (non-anonymized) dataset, certainty is 100%, given that all records naturally exist in this dataset. The same principle applies to the k-anonymized dataset, where we solely rely on generalization without employing suppression (i.e., removing outlier records in terms of QIDs). This ensures the inclusion of records from all users in the k-anonymized dataset.

Figure 12 displays the percentage values of certainty. As expected, in the k-anonymized dataset (referred to as "Anonymized" in the figure), the highest level of certainty is maintained at 100%. This value indicates that all records are retained in the released dataset, ensuring the utmost level of confidence in data presence and predictability. In contrast, the sampled anonymized datasets, particularly as the sampling percentages increase, introduce uncertainty and reduce the confidence in the presence of records within the dataset sent to data recipients. Striking a balance between anonymity concerns and the necessity for certainty in the outcomes suggests the importance of adjusting the sampling percentage to a value that isn't excessively low. This ensures a better trade-off between data anonymity and the reliability of the results.

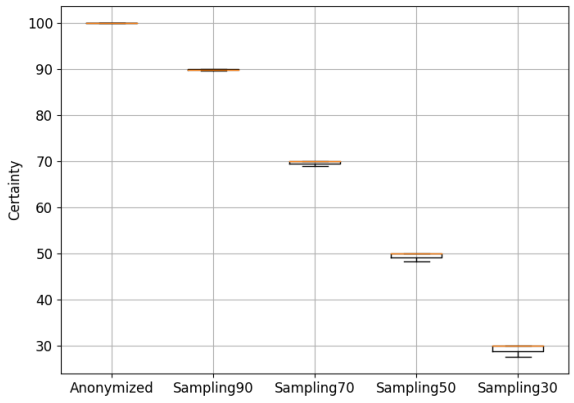


Figure 12: Certainty (k=250, Sampling percentage=90%, 70%, 50%, and 30%).

7 RELATED WORK

In this section, we discuss techniques commonly applied in practice [14] to safeguard privacy and anonymity within the context of medical data sharing.

Differential privacy (DP) is a technique that protects privacy by introducing noise to data, minimizing the impact of individual records on analysis results. Unlike other methods such as k-anonymity, ℓ-diversity, and t-closeness, DP doesn't rely on assumptions about attackers' knowledge [10]. Initially designed for interactive queries [10]—where data recipients can query the system interactively without access to the complete dataset—DP was later extended to non-interactive scenarios [11]. DP can employ noise addition in a local or global manner [26]. In local DP, noise is applied to individual data points independently, potentially causing more distortion than necessary due to a lack of consideration for the overall dataset. In contrast, global DP adds noise to the overall output, allowing characteristics of the dataset to be considered and generally leading to more accurate results. However, global DP, like k-anonymity, requires trust in the data aggregator from record owners to ensure privacy preservation. While effective against certain privacy threats, DP has drawbacks [21], including its inability to prevent all linkage attacks, introduced communication overhead, and potential hindrance of pattern detection in small populations or rare events due to added noise. Additionally, repeated queries on differentially private datasets can lead to privacy risks over time. Similarly to the challenges in k-anonymity, where selecting appropriate QIDs and finding a balanced k value is crucial, determining a suitable epsilon (ϵ) in DP presents a challenge due to the inevitable trade-off between privacy and utility.

Many data sharing protocols have been proposed, leveraging secure multi-party computation (SMPC) [34, 36, 41] or homomorphic encryption (HE) [31, 33, 42]. These protocols aim to ensure strong protection guarantees with minimal trust requirements. SMPC allows computations on distributed datasets without exposing raw data, ensuring each party retains control while defending against malicious attacks. Drawbacks of SMPC include computational overhead, increased communication complexity, and limited scalability [44]. On the other hand, HE enables computations on encrypted data, ensuring end-to-end confidentiality and eliminating the need for a central trusted authority. However, it faces challenges such as computational intensity, and slowing down processing, which limits its practicality [2]. Another common issue in data sharing protocols relying on SMPC or HE is their tendency to support specific or limited types of statistical analyses, hence limiting flexibility for researchers in conducting their studies.

Given the limitations of DP, SMPC, and HE, we assert that our protocol design offers superior flexibility to researchers compared to solutions based on any of these methods. We provide anonymized data to researchers without assuming the exact usage or nature of the studies, enhancing adaptability to diverse research needs. Furthermore, our protocol ensures better anonymity and data utility when compared to DP. The dataset anonymized by *Anonify* mitigates privacy risks associated with repeated queries, a concern inherent in differentially private datasets. *Anonify* depends on DPF, whose capability to support scalability without imposing high bandwidth and computational burdens has been demonstrated in the literature [13, 18]. That can position *Anonify* as a potentially superior option over solutions based on SMPC or HE in terms of scalability, bandwidth and computational efficiency.

8 CONCLUSION

Anonify is a decentralized anonymity protocol specifically designed for medical data donation. It offers users anonymous communication and data anonymity when donating their data without the need to trust a single entity. This is achieved through the utilization of a secret-sharing-based method called DPF, facilitating the anonymous sending of records, complemented by a broadcasting-based approach for anonymous data retrieval. Moreover, to mitigate data de-anonymization risks, we have employed k-anonymity and stratified sampling within a decentralized setting. Our comprehensive evaluation, encompassing various privacy and data utility metrics, demonstrates the effectiveness of *Anonify* in ensuring strong protection without compromising data utility.

ACKNOWLEDGMENTS

This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA - 390781972.

REFERENCES

[1] CC Aggarwal. 2014. Data classification: algorithms and applications.(1stedn).
[2] Bechir Alaya, Lamri Laouamer, and Nihel Msilini. 2020. Homomorphic encryption systems statement: Trends and challenges. *Computer Science Review* 36 (2020), 100235.
[3] Mina Alishahi and Nicola Zannone. 2021. Not a Free Lunch, But a Cheap One: On Classifiers Performance on Anonymized Datasets. In *Data and Applications Security and Privacy (DBSec)*. Springer, 237–258.
[4] Matthew Bietz, Kevin Patrick, and Cinnamon Bloss. 2019. Data donation as a model for citizen science health research. *Citizen Science: Theory and Practice* 4, 1 (2019).
[5] Elette Boyle, Niv Gilboa, and Yuval Ishai. 2016. Function Secret Sharing: Improvements and Extensions. (2016), 1292–1303. https://doi.org/10.1145/2976749.2978429
[6] Tânia Carvalho, Nuno Moniz, Pedro Faria, and Luís Antunes. 2022. Survey on privacy-preserving techniques for data publishing. *arXiv preprint*

arXiv:2201.08120 (2022).
[7] Henry Corrigan-Gibbs, Dan Boneh, and David Mazières. 2015. Riposte: An Anonymous Messaging System Handling Millions of Users. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*. IEEE Computer Society, 321–338. https://doi.org/10.1109/SP.2015.27
[8] D. Mentzer D. Oberle and G. Weber. 2020. Befragung zur Verträglichkeit der Impfstoffe gegen das neue Coronavirus (SARS-CoV-2) mittels Smartphone-App SafeVac 2.0. https://www.pei.de/SharedDocs/Downloads/EN/newsroom-en/pharmacovigilance-bulletin/single-articles/2020-safevac-app-en.pdf?__blob=publicationFile&v=3.
[9] George Danezis and Andrei Serjantov. 2004. Statistical disclosure or intersection attacks on anonymity systems. In *International Workshop on Information Hiding*. Springer, 293–308.
[10] Cynthia Dwork. 2006. Differential privacy. In *International colloquium on automata, languages, and programming*. Springer, 1–12.
[11] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*. Springer, 1–19.
[12] Khaled El Emam. 2013. *Guide to the de-identification of personal health information*. CRC Press.
[13] Saba Eskandarian, Henry Corrigan-Gibbs, Matei Zaharia, and Dan Boneh. 2021. Express: Lowering the cost of metadata-hiding communication with cryptographic privacy. In *30th USENIX Security Symposium (USENIX Security 21)*. 1775–1792.
[14] European Medicines Agency. 2017. Report on Data Anonymisation as a Key Enabler for Clinical Data Sharing. https://www.ema.europa.eu/en/documents/report/report-data-anonymisation-key-enabler-clinical-data-sharing_en.pdf. Accessed: February 12, 2025.
[15] Benjamin CM Fung, Ke Wang, Rui Chen, and Philip S Yu. 2010. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (Csur)* 42, 4 (2010), 1–53.
[16] Sarah Gaballah, Thanh Hoang Long Nguyen, Lamya Abdullah, Ephraim Zimmer, and Max Mühlhäuser. 2023. Mitigating Intersection Attacks in Anonymous Microblogging. In *Proceedings of the 18th International Conference on Availability, Reliability and Security*. 1–11.
[17] Sarah Abdelwahab Gaballah, Lamya Abdullah, Minh Tung Tran, Ephraim Zimmer, and Max Mühlhäuser. 2022. On the Effectiveness of Intersection Attacks in Anonymous Microblogging. In *Secure IT Systems - 27th Nordic Conference, NordSec 2022, Reykjavic, Iceland, November 30-December 2, 2022, Proceedings (Lecture Notes in Computer Science, Vol. 13700)*. Springer, 3–19. https://doi.org/10.1007/978-3-031-22295-5_1
[18] Sarah Abdelwahab Gaballah, Christoph Coijanovic, Thorsten Strufe, and Max Mühlhäuser. 2021. 2PPS - Publish/Subscribe with Provable Privacy. In *40th International Symposium on Reliable Distributed Systems, SRDS 2021, Chicago, IL, USA, September 20-23, 2021*. IEEE, 198–209. https://doi.org/10.1109/SRDS53918.2021.00028
[19] Nethanel Gelernter and Amir Herzberg. 2013. On the limits of provable anonymity. In *Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society*. 225–236.
[20] Aris Gkoulalas-Divanis, Grigorios Loukides, and Jimeng Sun. 2014. Publishing data from electronic health records while preserving privacy: A survey of algorithms. *Journal of biomedical informatics* 50 (2014), 4–19.
[21] Muneeb Ul Hassan, Mubashir Husain Rehmani, and Jinjun Chen. 2019. Differential privacy techniques for cyber physical systems: a survey. *IEEE Communications Surveys & Tutorials* 22, 1 (2019), 746–789.
[22] Hyukki Lee, Soohyung Kim, Jong Wook Kim, and Yon Dohn Chung. 2017. Utility-preserving anonymization for health data publishing. *BMC medical informatics and decision making* 17, 1 (2017), 1–12.
[23] Kristen LeFevre, David J DeWitt, and Raghu Ramakrishnan. 2006. Mondrian multidimensional k-anonymity. In *22nd International conference on data engineering (ICDE'06)*. IEEE, 25–25.
[24] Jun-Lin Lin and Meng-Cheng Wei. 2008. An efficient clustering method for k-anonymization. In *Proceedings of the 2008 international workshop on Privacy and anonymity in information society*. 46–50.
[25] D. Oberle, D. Mentzer, and G. Weber. 2020. https://www.pei.de/SharedDocs/Downloads/EN/newsroom-en/pharmacovigilance-bulletin/single-articles/2020-safevac-app-en.pdf?__blob=publicationFile&v=3. Accessed: 2022-10-03.
[26] Iyiola E Olatunji, Jens Rauch, Matthias Katzensteiner, and Megha Khosla. 2022. A review of anonymization for healthcare data. *Big data* (2022).
[27] European Parliament. 2016. Regulation (EU) 2016/679 of the European Parliament and of the council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:32016R0679 (Accessed 02-February-2021).
[28] Van L Parsons. 2014. Stratified sampling. *Wiley StatsRef: Statistics Reference Online* (2014), 1–11.
[29] Fabian Prasser and Florian Kohlmayer. 2015. Putting statistical disclosure control into practice: The ARX data anonymization tool. *Medical data privacy handbook* (2015), 111–148.
[30] RKI. 2019. Corona Data Donation Project. https://corona-datenspende.de/science/en/.
[31] Bharath K Samanthula, Gerry Howser, Yousef Elmehdwi, and Sanjay Madria. 2012. An efficient and secure data sharing framework using homomorphic encryption in the cloud. In *Proceedings of the 1st International Workshop on Cloud Intelligence*. 1–8.
[32] Pierangela Samarati. 2001. Protecting respondents identities in microdata release. *IEEE transactions on Knowledge and Data Engineering* 13, 6 (2001), 1010–1027.
[33] Hossein Shafagh, Anwar Hithnawi, Lukas Burkhalter, Pascal Fischli, and Simon Duquennoy. 2017. Secure sharing of partially homomorphic encrypted iot data. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 1–14.
[34] Haoyi Shi, Chao Jiang, Wenrui Dai, Xiaoqian Jiang, Yuzhe Tang, Lucila Ohno-Machado, and Shuang Wang. 2016. Secure multi-pArty computation grid LOgistic REgression (SMAC-GLORE). *BMC medical informatics and decision making* 16 (2016), 175–187.
[35] Joanna Sleigh. 2018. Experiences of donating personal data to mental health research: an explorative anthropological study. *Biomedical Informatics Insights* 10 (2018), 1178222618785131.
[36] Haris Smajlović, Ariya Shajii, Bonnie Berger, Hyunghoon Cho, and Ibrahim Numanagić. 2023. Sequre: a high-performance framework for secure multiparty computation enables biomedical data sharing. *Genome Biology* 24, 1 (2023), 5.
[37] Latanya Sweeney. 2002. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10, 05 (2002), 571–588.
[38] Latanya Sweeney. 2002. k-anonymity: A model for protecting privacy. *International journal of uncertainty, fuzziness and knowledge-based systems* 10, 05 (2002), 557–570.
[39] Jenno Verdonck, Kevin De Boeck, Michiel Willocx, Jorn Lapon, and Vincent Naessens. 2023. A hybrid anonymization pipeline to improve the privacy-utility balance in sensitive datasets for ML purposes. In *the 18th International Conference on Availability, Reliability and Security*. 1–11.
[40] Torsten H Voigt, Verena Holtz, Emilia Niemiec, Heidi C Howard, Anna Middleton, and Barbara Prainsack. 2020. Willingness to donate genomic and other medical data: results from Germany. *European Journal of Human Genetics* 28, 8 (2020), 1000–1009.
[41] Felix Nikolaus Wirth, Tobias Kussel, Armin Müller, Kay Hamacher, and Fabian Prasser. 2022. EasySMPC: a simple but powerful no-code tool for practical secure multiparty computation. *BMC bioinformatics* 23, 1 (2022), 531.
[42] Alexander Wood, Kayvan Najarian, and Delaram Kahrobaei. 2020. Homomorphic encryption for machine learning in medicine and bioinformatics. *ACM Computing Surveys (CSUR)* 53, 4 (2020), 1–35.
[43] Jian Xu, Wei Wang, Jian Pei, Xiaoyuan Wang, Baile Shi, and Ada Wai-Chee Fu. 2006. Utility-based anonymization using local recoding. In *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data*. ACM, 785–790. https://doi.org/10.1145/1150402.1150504
[44] Chuan Zhao, Shengnan Zhao, Minghao Zhao, Zhenxiang Chen, Chong-Zhi Gao, Hongwei Li, and Yu-an Tan. 2019. Secure multi-party computation: theory, practice and applications. *Information Sciences* 476 (2019), 357–372.

A APPENDIX

The following figures show the eight machine learning classifiers' performance in terms of Recall, Precision, and F1-score, respectively.

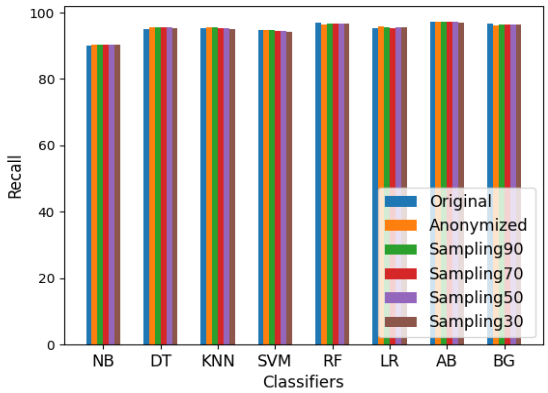


Figure 13: Recall (k=250, Sampling percentage=90%, 70%, 50%, and 30%).

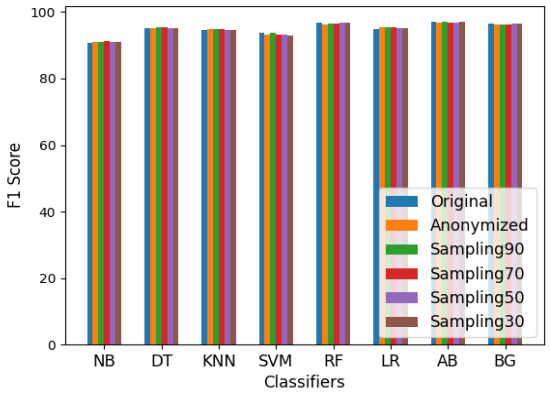


Figure 15: F1 Score (k=250, Sampling percentage=90%, 70%, 50%, and 30%).

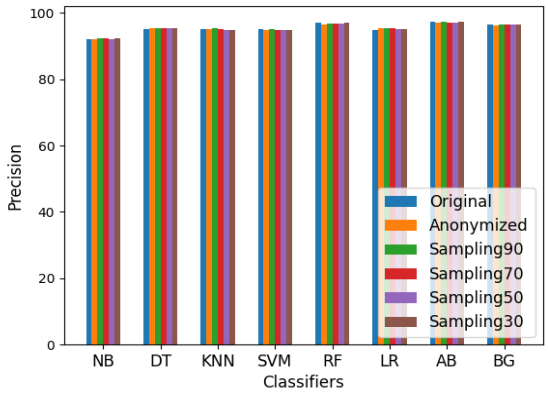


Figure 14: Precision (k=250, Sampling percentage=90%, 70%, 50%, and 30%).

P3

ON THE EFFECTIVENESS OF INTERSECTION ATTACKS IN ANONYMOUS MICROBLOGGING

This chapter was first published as

Sarah Abdelwahab Gaballah, Lamya Abdullah, Minh Tung Tran, Ephraim Zimmer, and Max Mühlhäuser. "On the Effectiveness of Intersection Attacks in Anonymous Microblogging." In 27th Nordic Conference on Secure IT Systems (NordSec), pp. 3-19. Springer, 2022.

Contribution Statement: I led the idea generation, conceptualization, and development of the proposed work, as well as the experiment design, data analysis, and writing of the publication. All co-authors helped with critiques and comments on the concept design and participated in the creation of the publication.

On the Effectiveness of Intersection Attacks in Anonymous Microblogging

Sarah Abdelwahab Gaballah, Lamya Abdullah, Minh Tung Tran, Ephraim Zimmer, and Max Mühlhäuser

Telecooperation Lab (TK), Technical University of Darmstadt, Darmstadt, Germany
{gaballah, abdullah, max}@tk.tu-darmstadt.de,
minhtung.tran@stud.tu-darmstadt.de, zimmer@privacy-trust.tu-darmstadt.de

Abstract. Intersection attacks, which are popular traffic analysis attacks, have been extensively studied in anonymous point-to-point communication scenarios. These attacks are also known to be challenging threats to anonymous group communication, e.g., microblogging. However, it remains unclear how powerful these attacks can be, especially when considering realistic user communication behavior. In this paper, we study the effectiveness of intersection attacks on anonymous microblogging systems utilizing Twitter and Reddit datasets. Our findings show that the attacks are effective regardless of whether users post their messages under pseudonyms or publish them to topics without attaching identifiers. Additionally, we observed that attacks are feasible under certain settings despite increasing userbase size, communication rounds' length, cover traffic, or traffic delays.

Keywords: Anonymous Communication · Traffic analysis · Intersection attacks · Microblogging

1 Introduction

Microblogging is one of the most popular forms of online social networking that attracts millions of users. Twitter, for example, is one of the leading microblogging services. It had approximately 290.5 million active users worldwide, monthly, in 2019, with a projected increase to over 340 million users by 2024 [9]. All the known microblogging services are based on a centralized architecture, that enables those systems to know everything about users' messages and interests [28]. These services may collect data about their users and sell or reveal this information to third parties such as governments. In fact, many services already do this, for example, Facebook said that it has produced data for 88% of the U.S. government requests [24]. This type of data disclosure could endanger many users, including political dissidents, human rights activists, and people who want to share sensitive information (e.g., about health problems, or sexual harassment experience, etc). Therefore, many Anonymous Communication Systems (ACSs) have been proposed over recent years in order to conceal user's interests from the microblogging service providers, other users, and even a global adversary

H. P. Reiser and M. Kyas (Eds.): NordSec 2022, LNCS 13700, pp. 3–19, 2022.
https://doi.org/10.1007/978-3-031-22295-5_1

(e.g., an internet service provider, a government authority, or an intelligence agency) who can monitor the network communication, by means of hiding their metadata (e.g., who is communicating with whom, when, and how frequently they communicate). Some of these systems are specifically focused and designed towards social networking scenarios such as microblogging [1, 4, 8, 10, 15, 16, 20].

The vast majority of the existing ACSs are vulnerable to traffic analysis attacks [6]. Intersection attacks are one of the most common and powerful types of traffic analysis attacks [27]. This type of attack takes advantage of the change in the set of users participating in the system over time. An ACS initially might ensure that a user is not identifiable within a set of other users, which is called an anonymity set [21]. However, changes in the communication behaviors of users (e.g., online and offline time of participating users) will evolve and add up to further information for the adversary in order to reduce the anonymity set or even single out (i.e., de-anonymize) specific users. A potentially large anonymity set can be reduced over time just by monitoring, storing, and repeatedly intersecting the online status of participating users when new messages are exchanged. To deduce with absolute certainty that two users are communicating, the adversary usually needs to launch the attack for a long time [2]. While intersection attacks are deterministic, there is a probabilistic version. It is called statistical disclosure attacks, which enables an adversary to estimate the likelihood that a targeted sender was communicating with a specific recipient [12].

Intersection attacks are commonly used to link sender and recipient in anonymous point-to-point communication settings. These attacks, however, can also be applied to anonymous group communication scenarios, such as anonymous microblogging [27]. Because user messages are publicly published in the microblogging scenario, an adversary can leverage the use of the intersection attacks to link users to their published messages or topics of interest.

The existing literature has studied intersection attacks, statistical disclosure attacks, and suitable mitigation approaches to those attacks in ACSs extensively [2, 7, 12, 14, 23, 25, 27]. However, we identified several pitfalls that render those works incomplete under realistic communication scenarios. First, user communication behavior often is assumed to follow a Poisson distribution [17]. On the contrary, by utilizing real-world data collection, it has already been shown that realistic communication behavior does, in fact, not follow such a distribution [17]. The communication behavior, however, has a huge impact on the effectiveness of intersection attacks as well as on mitigation measures, as will be shown in the remainder of this paper. Which is why common assumptions about the user communication behavior must be reconsidered. Second, proposals of ACSs, such as [1, 3, 10], consider constant user participation, i.e., the requirement of users to be always online and sending messages to the system, as the only way to protect against intersection attacks. Another approach proposed in [12, 27] is grouping users into anonymity sets and only allows all users to join the system at the same time and having the same sending rate. Yet, constant user participation, homogeneous user joining, and message sending does not reflect realistic user communication behavior either, and enforcing it would

significantly reduce the practicability of ACSs. Third, common mitigation techniques proposed against intersection attacks suggest either an increase in the size of the userbase as the anonymity set, a random delay of sending messages to the communication system, or utilizing cover traffic to hide real message sending. However, it has been demonstrated that those mitigation measures cannot provide long-term protection against intersection attacks [2,11]. Still, a detailed investigation of these mitigation effects and their usefulness in realistic communication settings and especially in microblogging scenarios are missing.

To the best of our knowledge, no previous work has provided a *practical investigation* which answers the following questions:

- How effective are the intersection attacks on different anonymous microblogging scenarios? Especially, when realistic communication settings are considered.
- What impact does realistic user communication behavior (e.g., sending rates) have on these attacks?
- To what extent can common mitigation measures against intersection attacks, more specifically the increasing in cover traffic, delay, and userbase, facilitate the attack mitigation under realistic microblogging scenarios?

We believe that answering these questions is critical to understand how and under what conditions intersection attacks are powerful. Additionally, this understanding can facilitate and enhance the development of effective mitigation measures to protect against intersection attacks in practical ACSs. In this paper, we address the aforementioned questions by investigating intersection attacks on two different messaging patterns: pseudonym-based and topic-based messaging of anonymous microblogging. We use real-world datasets from Twitter and Reddit to simulate realistic user communication behavior. We launch the attack to reveal the identity of the publishing user (i.e., de-anonymize the user), hence the posts and messages can be linked to the sending user. Using an intersection attack for the other direction, i.e., to discover which topic a user subscribed to is beyond our scope.[1]

This paper is organized as follows: The assumptions, anonymous microblogging messaging patterns, and threat model are all introduced in Section 2. Then, in Section 3, we present our intersection attack for each messaging pattern. Following that, in Section 4, we discuss the evaluation results of our experiments. Finally, Section 5 concludes the paper and presents some points for future work.

2 Design and Assumptions

In this work, we consider an anonymous microblogging system that has a set of users $U = \{u_1, u_2, ...u_n\}$. Each user, $u_i \in U$, can independently decide how

[1] It is noteworthy that a solution for this intersection attack direction has been addressed already by utilising *broadcasting* of published messages, i.e., sending every published message to all users, as seen in [1,4,5]. Nonetheless, broadcasting imposes a high communication overhead on users, which makes it an inefficient solution. Thus, more research in this area is clearly needed.

many messages to send, and when to send them. Thus, there are no agreements on communication behavior between users, which is closest to commonly used existing non-anonymous microblogging systems. All messages transmitted between users and the system are encrypted and padded to the same length i.e., they appear indistinguishable to any external observer. The users do not include any personally identifiable information (e.g., real names or addresses) in their messages, as the messages are published publicly on the system. The communication in the system is assumed to proceed in rounds, $R = \{r_1, r_2, ...r_w\}$, and u_i is *online* in a round $r_x \in R$ when she sends messages during this round. The system publishes all messages at the end of the round. It also can support special anonymity features like cover traffic and the delay of messages.

2.1 Messaging Patterns

To investigate the effectiveness of intersection attacks on different anonymous microblogging scenarios, we consider two major microblogging messaging patterns which differ based on how users publish their messages.

Pseudonym-based messaging pattern. Every u_i uses a *pseudonym* (i.e., a fictitious username), $p' \in P = \{p_1, p_2, ...p_m\}$, in order to publish her messages in the system, see Fig. 1(a). The pseudonyms must not contain any personal information about the users. The system is responsible for ensuring *unlinkability* between u_i and p'. Any user of the system can read the content published under every pseudonym but cannot realize the real identity of the pseudonym's owner. In this pattern, we assume a one-to-one relationship between the sets of users U and pseudonyms P, which means that each user $u_i \in U$ has only one pseudonym $p' \in P$ and vice versa. The pseudonym is considered *online* during round r_x only when messages are published under this pseudonym during r_x.

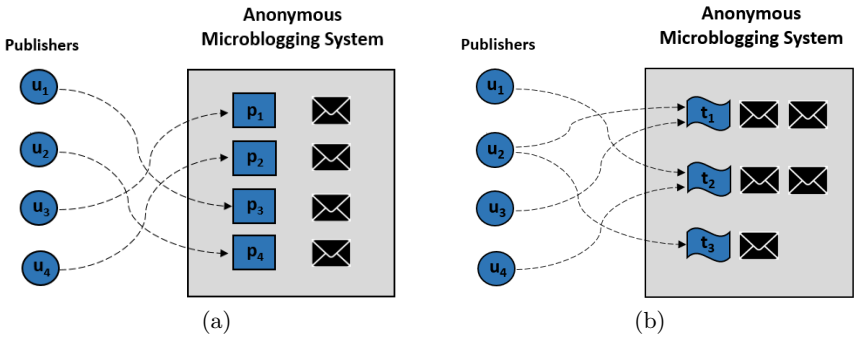


Fig. 1. The publishing process in (a) Pseudonym-based messaging pattern and (b) Topic-based messaging pattern.

Topic-based messaging pattern. Users follow the *topic-based publish/subscribe* (pub/sub) paradigm. In this pattern, the users publish their messages to topics i.e., a user u_i who posts a message to a topic $t' \in T = \{t_1, t_2, ...t_q\}$ is called a publisher, see Fig. 1(b). As the topics are the focus, publishers do not need to have any kind of identity (e.g., pseudonyms) on the anonymous microblogging system. Ensuring unlinkability between u_i and t' is assumed to be assured by the system. Any user of the system can read every message published to every topic but cannot know anything about who publishes these messages. In this pattern, the users U and Topics T have a many-to-many relationship, which means that t' can have many publishers and u_i can publish to multiple topics. Additionally, we assume a topic to be *active* during round r_x when it receives new messages in r_x. Furthermore, any user can send messages to multiple topics in the same round.

2.2 Threat Model

We assume a global passive adversary $\mathcal{A}$ who monitors the whole communication during the set of rounds R and can learn who participates in each round and how many messages each participant sends. $\mathcal{A}$ does not have any means to interrupt the traffic by dropping or altering data packets, nor does $\mathcal{A}$ gain intelligence about the actual content of the encrypted messages during transmission. It cannot collude with the anonymous system, but it can corrupt an arbitrary number of users. Moreover, it can read all published messages every round.

2.3 Delay

The messages can be arbitrarily delayed on the system for a number of rounds ($\leq d$ rounds) to prevent $\mathcal{A}$ from correlating the incoming messages—sent by the users to the system—with the published ones. $\mathcal{A}$ can realize the maximum allowed delay (d), but it cannot learn the exact number of rounds for which a message will be delayed. To deal with this issue, when a message is sent in r_x, $\mathcal{A}$ shall observe the published content during $r_x, r_{x+1}, \cdots, r_{x+d-1}$ as the message must be published in one of these rounds. Therefore, $\mathcal{A}$ will treat these consecutive rounds as one joint big round.

2.4 Cover Traffic

Users can produce cover traffic (also referred to as *dummy messages*) to prevent $\mathcal{A}$ from discovering when and how many real messages are sent. Since the system does not publish cover messages, $\mathcal{A}$ can detect the presence of cover traffic if the total number of sent messages by users is larger than the total number of eventually published messages. Nonetheless, it is not possible to determine the exact number of real messages sent by each user. $\mathcal{A}$ can only be certain that the user's real messages are less than or equal to the number of messages sent by her.

3 Attacks

In this section, we explain how the adversary can de-anonymize users by employing an intersection attack, considering the two anonymous microblogging messaging patterns mentioned in Section 2.1.

3.1 User-Pseudonym Linking

This attack targets anonymous microblogging systems that are based on the pseudonym-based messaging pattern. The goal is to identify the pseudonym p' of a user u_i to learn what she publishes. We do not consider a statistical disclosure attack as the aim is to study what the adversary can learn with *absolute certainty*. Thus, we consider $\mathcal{A}$ as being *successful* only when it can narrow down the u_i's list of potential pseudonyms to only p'.

The *naive approach* for launching the attack (i.e., link u_i to p') is described as follows: When u_i is *online* for the first time in round r_x, $\mathcal{A}$ creates the set of potential pseudonyms, P_{u_i}, which constitutes the anonymity set of u_i at this point. P_{u_i} contains every *online* pseudonym p_j in r_x that meets the following two requirements:

- p_j is *online* for the first time, i.e., messages are published under this pseudonym for the first time.
- $M_{p_j} = M_{u_i}$, where M_{p_j} is the number of messages published under p_j, and M_{u_i} is the number of messages sent by u_i.

In every subsequent round $(r_{x+1}, r_{x+2}, \cdots)$, if u_i sends a message, any pseudonym $p_l \in P_{u_i}$ that does not publish new messages, shall be removed from P_{u_i}. When the size of P_{u_i} drops to only one, it means that the pseudonym p' of u_i is identified. However, this naive approach does not consider the following issues:

- u_i may send only dummy messages either in her first round (r_x) or in subsequent rounds. In this case, u_i will be *online* but her pseudonym p' will not be *online* as the dummy messages are not published on the system.
- u_i may send both real and dummy messages either in r_x or in subsequent rounds. In this case, p' will have published fewer messages than what u_i has sent (i.e., $M_{p'} < M_{u_i}$).

To overcome these issues and perform a more powerful intersection attack for linking a user to her pseudonym, we consider the following approach for our investigations. It is based on the observation that the attack can be more efficient if $\mathcal{A}$ does not only consider the *online* rounds of u_i but also her *offline* rounds. In other words, $\mathcal{A}$ can track pseudonyms in P_{u_i} which are *online* while u_i is *offline* and excludes them from P_{u_i} which might lead to faster convergence:

Step 1. Creating the initial anonymity set.

Let P_{u_i} be the set of potential pseudonyms for u_i (anonymity set) . Initially, it includes every p_j that publish messages in r_x for the first time while at the same time u_i is online for the first time as well. Additionally, p_j has published the same or a fewer amount of messages compared to the number of messages sent by u_i, i.e., $M_{p_j} \leq M_{u_i}$.

Step 2. Verifying whether p' is in P_{u_i}.

$\mathcal{A}$ compares the total number of *online* users to the total number of *online* pseudonyms during r_x. If they are equal, it means all online users published real messages, so P_{u_i} definitely includes p'. In such a case, a flag f is set to *true*, which indicates that no new pseudonyms can be added to P_{u_i}.

Step 3. Updating P_{u_i} during every subsequent round.

(a) Every p_l that fulfills one of the following conditions will be removed from P_{u_i}:
   - it is *online* while u_i is *offline*.
   - $M_{p_l} > M_{u_i}$.
   - it is *offline* while u_i is *online* and the number of online users is the same as the number of *online* pseudonyms, i.e., p' is definitely *online* in this particular round. (When this happens, the flag f must be set to *true*.)

(b) If u_i is *online* and f is *false*, $\mathcal{A}$ adds to P_{u_i} every pseudonym p_j that is online for the first time, and $M_{p_j} \leq M_{u_i}$.

Step 4. Repeat Step 3 until: the flag f is *true* and P_{u_i} contains only one pseudonym (p').

Fig. 2 shows a simple example of the proposed method. In this example, we assume a group of four users $U = \{u_1, u_2, u_3, u_4\}$ publishing during four rounds $R = \{r_1, r_2, r_3, r_4\}$, and the goal is to determine each user's pseudonym. $\mathcal{A}$ can learn the following over each round :

- r_1: $P_{u_1} = \{p_2, p_3\}$, $P_{u_2} = \{p_2, p_3\}$, $P_{u_3} = \{p_2, p_3\}$
- r_2: $P_{u_1} = \{p_1, p_3, p_4\}$, $P_{u_2} = \{p_2\}$, $P_{u_3} = \{p_1, p_3, p_4\}$, $P_{u_4} = \{p_1, p_4\}$
- r_3: $P_{u_1} = \{p_1, p_3, p_4\}$, $P_{u_2} = \{p_2\}$, $P_{u_3} = \{p_1, p_3\}$, $P_{u_4} = \{p_1, p_4\}$
- r_4: $P_{u_1} = \{p_1\}$, $P_{u_2} = \{p_2\}$, $P_{u_3} = \{p_3\}$, $P_{u_4} = \{p_4\}$

Intuitively, the more rounds that are observed, the more information is gathered; thus, the likelihood of a pseudonym being exposed increases. As we discussed in Section 2.2, $\mathcal{A}$ does not know the actual content of the messages sent, so it cannot distinguish between the cover messages and the real ones.

3.2 User-Topic Linking

This attack targets anonymous microblogging systems that use a topic-based messaging pattern. The adversary's goal is to identify the topic(s) on which u_i is publishing extensively. In this attack, a ranking-based approach [18] is utilized for our investigations, where $\mathcal{A}$ keeps a list of potential interest topics for user u_i, let's call it T_{u_i}. Each topic $t_k \in T_{u_i}$ is assigned a score (initially zero). The topic(s) with the highest score is most likely to be the one with the most posts by the user. This attack is especially effective when the user has been immersed in a specific topic for a long time. The scores of topics can be calculated by $\mathcal{A}$ using the methods outlined below.

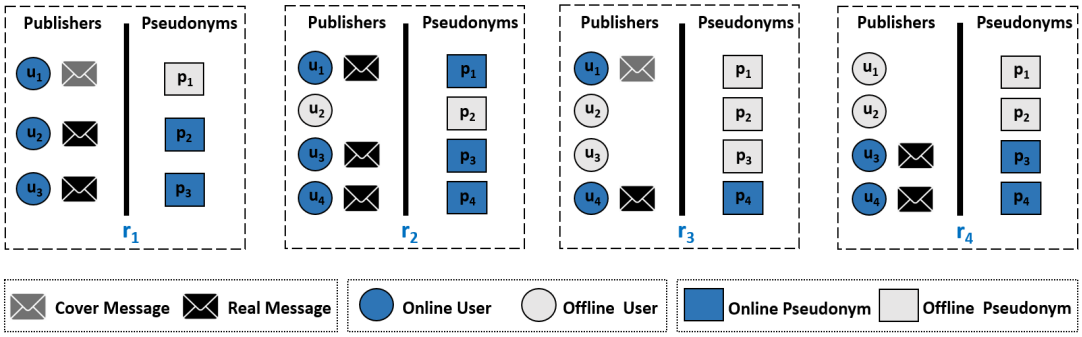


Fig. 2. Example of the User-Pseudonym Linking Attack

Method 1: In every round r_x in which u_i is *online*, $\mathcal{A}$ adds to T_{u_i} any topic that is *active* for the first time. Then, it updates the scores of topics that are in T_{u_i} by taking the following actions:

- increasing the score of each $t_k \in T_{u_i}$ by a number a if it is *active* in r_x.
- decreasing the score of each $t_k \in T_{u_i}$ by a number b if it is *inactive* in r_x.

$\mathcal{A}$ cannot realize the topic(s) on which u_i publishes messages. Therefore, it considers every *active* topic if u_i is *online* as a potential interest and it increases the score of this topic. To expand the score gap between interesting topics and not-interesting ones, $\mathcal{A}$ decreases the score of topics that are *inactive* when u_i is *online*. It does not exclude these topics, as u_i may post messages on them during subsequent rounds.

To maintain high scores for the topics that are in fact interesting to u_i, but that are not posted to by u_i in every round, i.e., they are sometimes *inactive* even though u_i is *online*, the increasing number a should be greater than the decreasing number b.

Fig. 3 shows a simple example for using Method 1 to update the scores of the topics over rounds, assuming u_i is *online* during the rounds $R = \{r_1, r_2, r_3, r_4\}$. In this example, at the end of the four rounds, t_4 is assumed to be the most likely interesting topic for u_i because it has the highest score. For simplicity, a and b have been set to one, so the increasing and decreasing of each topic's score is done by one. Later, in the evaluation, we discuss further the increasing and decreasing numbers.

Method 2: Some topics are so popular that they are *active* throughout the majority of the rounds. By just observing rounds in which u_i is *online*, these topics will get high scores in T_{u_i} even if they are not of any interest to the user. To tackle this issue, in addition to updating the topics' scores during rounds in which u_i is *online* (similar to Method 1), $\mathcal{A}$ can do the following:

- for every round in which u_i is *offline*, $\mathcal{A}$ adds to T_{u_i} any topic that is *active* for the first time, and decreases the score of each $t_k \in T_{u_i}$ by a number b if it is *active*.

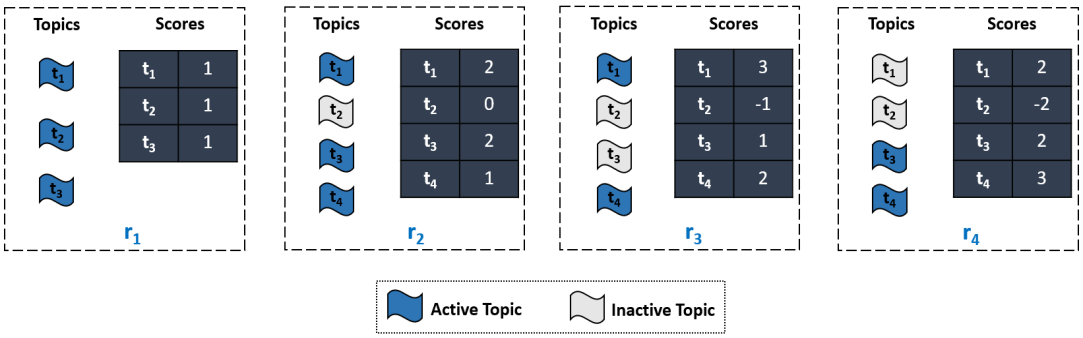


Fig. 3. Example of the User-Topic Linking Attack

That enables excluding irrelevant highly *active* topics while retaining relevant highly *active* topics as top-ranked topics in T_{u_i}.

Method 3: u_i may be interested in several topics that are not widely popular. Since u_i may not publish on all of these topics every time she goes *online*, some of them may be *inactive* when she is *online*. Reducing the scores of these topics every time u_i is *online* can result in a significant drop in their scores. Thus, in order to keep high scores for these topics, in this method, $\mathcal{A}$ does not reduce the score of every *inactive* $t_k \in T_{u_i}$ in the rounds in which u_i is *online*. The steps of this method, executed every round regardless whether u_i is *online* or not, are as follows:

- adding to T_{u_i} any topic that is *active* for the first time.
- increasing the score of each $t_k \in T_{u_i}$ by a number a if it is *active* when u_i is *online*.
- decreasing the score of each $t_k \in T_{u_i}$ by a number b if it is *active* when u_i is *offline*.

4 Evaluation

In this section, we measure the effectiveness of the presented intersection attacks for linking a user to a pseudonym and/or topic on anonymous microblogging. The communication is simulated using two real-world datasets collected from two popular microblogging platforms, Twitter and Reddit. Users' messages are assigned to communication rounds based on their timestamps in the datasets. All rounds have the same length of time. We tested three round length values: 30 minutes (1,800 seconds), 1 hour (3,600 seconds), and 2 hours (7,200 seconds), where an increase in the round length usually means an increase in the number of users participating in the round, i.e., anonymity set size. To simulate the adversary $\mathcal{A}$, a *logger* was implemented to record all traffic generated by users and all messages posted on the anonymous microblogging system during rounds.

The data gathered by the *logger* serves as input for an *analyzer*, which executes the attacks. Our simulation prototype is implemented using Java and Python. We conducted the experiments on a machine equipped with a 16-core Intel Xeon E5-2640 v2 processor and 64 GB of RAM.

Datasets. The first dataset is a collection of records extracted from Twitter over the course of the entire month of November 2012 [19] [13]. This dataset contains 22,534,846 tweets, 6,914,561 users, and 3,379,976 topics, referred to as hashtags. The second dataset is collected by us from Reddit for the entire month of October 2021. This dataset contains posts and comments from 1,638,157 different users and 3,403 different topics, referred to as subreddits. Both datasets include a timestamp, user id, and topic (hashtag/subreddit) at each record.

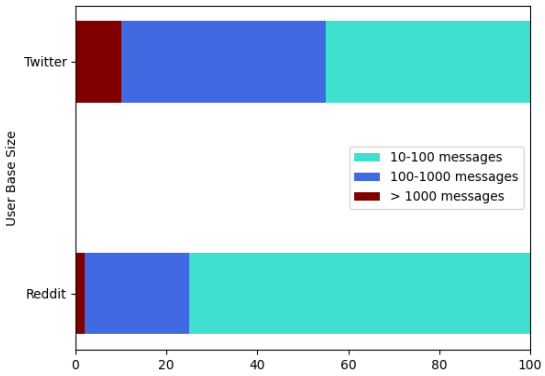


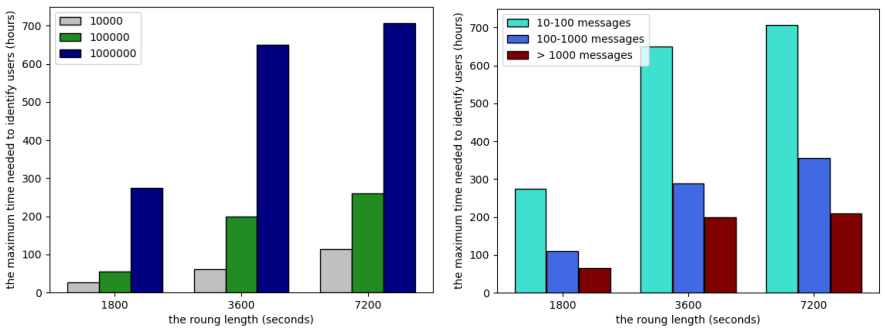
Fig. 4. The distribution of users based on sending rates (userbase size = 1 million)

Users. To investigate the impact of the number of users on the performance of the intersection attacks, in our experiments, we tested userbases of different sizes: 10,000 , 100,000 , and 1,000,000 . Each userbase is created by choosing users randomly and independently from the datasets. To study the influence of user communication behavior on the attacks, we looked into the relationship between how many messages a user sends (user's sending rates) and the vulnerability to the attack. For that, we focused on three groups of sending rates in particular: $(10 - 100)$ messages, $(100 - 1{,}000)$ messages, and $(> 1{,}000)$ messages. Fig. 4 shows the percentage of users who belong to each group in the userbase of one million users. The shown distribution was found to be nearly the same in 10,000 and 100,000 userbases. As illustrated in Fig. 4, most Reddit users sent between 10 and 100 messages during the observed month, which is different from the Twitter dataset. The difference in sending rates between users on Twitter and Reddit is expected, given that the two platforms have different service models.

Cover Traffic and Delay. To study the implications of using cover traffic on the effectiveness of the intersection attacks, each user generates a fixed number of dummy messages to hide every single real message (this is the same cover traffic generation approach used in [22]). The generated dummy messages are sent in random rounds to create noise in the user's sending rate. We tested three different cover-to-real message ratios: 1 :1 , 5 :1 , and 10 :1 . Similarly, to study the effectiveness of the delay on the intersection attacks, we evaluated three values for the maximum delay d in terms of number of rounds: one round, three rounds, and five rounds. Messages are delayed for an arbitrary number of rounds up to at most d rounds.

4.1 User-Pseudonym Linking

To evaluate the effectiveness of the user-pseudonym linking attack, we computed the maximum amount of time required to link a user to her pseudonym (de-anonymization), as shown in Fig. 5(a) and Fig. 5(b). For instance, $\mathcal{A}$ needs a maximum of 200 rounds to learn the pseudonym of any user—regardless of the sending rate—when the userbase size is 100,000 users and the round length is 3,600 seconds, see Fig. 5(a). While as illustrated in Fig. 5(b), when the system has one million users and the round length is 7,200 seconds, then $\mathcal{A}$ needs a maximum of 210 rounds to learn the pseudonym of any user who has sent more than 1,000 messages. As depicted in Fig. 5(a) and Fig. 5(b), the time needed for the de-anonymization increases when the round length, the number of users, or the sending rates increase. Nonetheless, having a large round length or a large user base still does not provide long-term protection, especially for users with high sending rates.



(a) (b)

Fig. 5. The maximum time needed to de-anonymize users in the Twitter dataset. (a) studying the impact of various userbase sizes, where the userbase includes users of different sending rates, (b) studying the impact of sending rates (userbase size = 1 million).

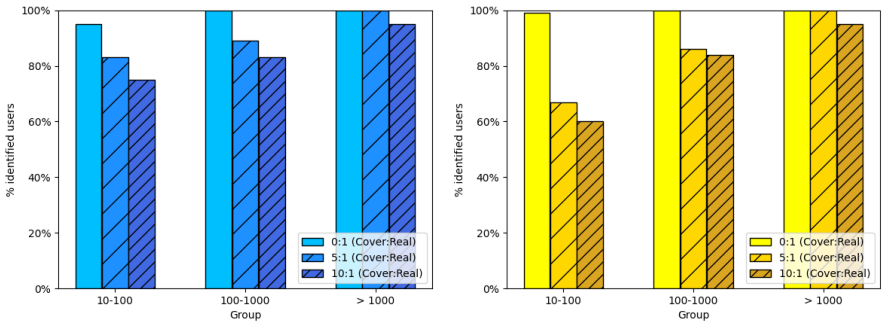
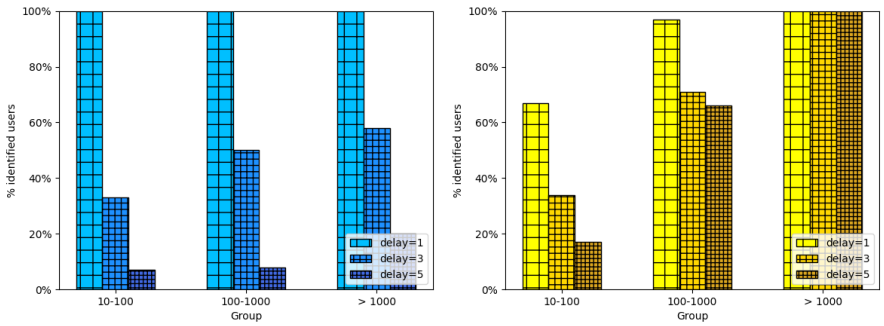


Fig. 6. The impact of cover traffic on the number of de-anonymized users (userbase size = 1 million, round length = 1 hour). (a) Twitter, (b) Reddit.

In Fig. 6(a) and Fig. 6(b), we show the impact of using cover traffic on the number of identified users in the Twitter and Reddit datasets, respectively. According to our findings, sending random cover traffic increases the number of required observed rounds. However, it can only slightly reduce the number of de-anonymized users by the end of the observation period. For example, if users randomly send 10 dummy messages for every real message (i.e., ratio 10:1)—that definitely leads to high bandwidth overhead—, the adversary still can identify pseudonyms of over 70 %, 80 %, and 90 % of users who send $(10 - 100)$, $(100 - 1{,}000)$, and $(> 1{,}000)$ messages, respectively, see Fig. 6(a). We think that the main reason behind the ineffectiveness of cover traffic in protecting the users is the randomness in generating and sending the dummy messages, which seems incapable of creating anonymity sets that can provide long-term protection for the users' pseudonyms.

In Fig. 7(a) and Fig. 6(b), we illustrate how delaying messages can help in degrading the effectiveness of the user-pseudonym linking attack. Since the adversary cannot learn the exact number of rounds for which a message has been delayed, it treats every message as if it has been postponed by d rounds. That results in very large anonymity sets, hence, it reduces the attack performance. Using the delay is more powerful on Twitter than on Reddit, especially for users with sending rate of more than 100 messages. That is because nearly 60 % of users in the Twitter dataset post more than 100 messages, whereas there are only about 25 % of users in the Reddit dataset who have similar sending rates, see Fig. 4. In the Reddit dataset, for example, there are only five users who have sent more than 1,000 messages. The communication behaviors of these five users are also noticeably different, making it difficult to conceal each one's behavior using delay. Nonetheless, it appears that cover traffic is more effective in protecting these users as shown in Fig. 6(b).

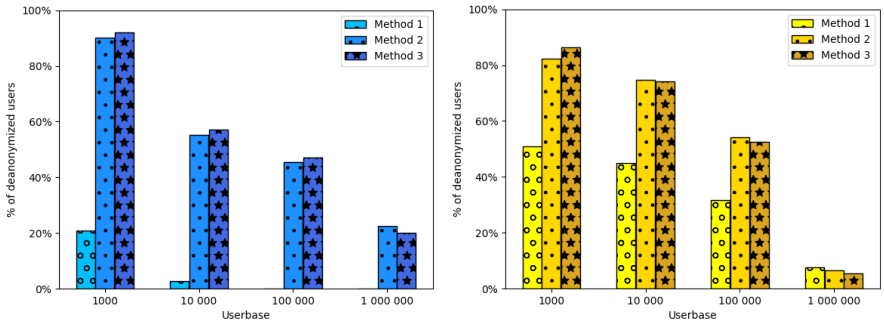


(a) (b)

Fig. 7. The impact of delay on the number of de-anonymized users (userbase size = 1 million, and round length = 1 hour). (a) Twitter, (b) Reddit.

4.2 User-Topic Linking

The increasing and decreasing numbers (a and b), described in Section 3.2, were tested using several values. We found that in the Twitter dataset, the best results can be produced for all userbases when $a = 5$ and $b = 1$. While in the Reddit dataset, the best values are $a = 7$ and $b = 1$.



(a) (b)

Fig. 8. The impact of increasing the userbase size on the effectiveness of three methods in linking users to topics (sending rate group is (100 − 1,000), and round length = 1 hour). (a) Twitter (a=5, b=1), (b) Reddit (a=7, b=1).

In Fig. 8(a) and Fig. 8(b), the effectiveness of the user-topic linking attack is studied using each of the three methods of computing topic scores, described in Section 3.2. As demonstrated in both figures, any increase in the userbase leads to a significant decrease in the number of de-anonymized users. Furthermore,

predictably, the attack is greatly influenced by the users' sending rates, i.e., users posting more messages are much more vulnerable to the attack. In the figures, we show only the results of users who sent $(100 - 1{,}000)$ messages.

The second and third methods produce significantly better results than the first method, see Fig. 8(a) and Fig. 8(b). Since the irrelevant topics are filtered out in the second and third methods by decreasing the scores of topics that are *active* when the user is *offline*. The third method appears to behave similarly to the second, implying that lowering the scores of *inactive* topics when the user is *online* has little effect on the results.

The first method seems to be far more effective on Reddit than on Twitter, implying that updating the scores of topics during users' *offline* rounds has less impact on Reddit than on Twitter. That is due to differences in the two datasets; e.g., Reddit has a smaller number of high popular topics compared to Twitter.

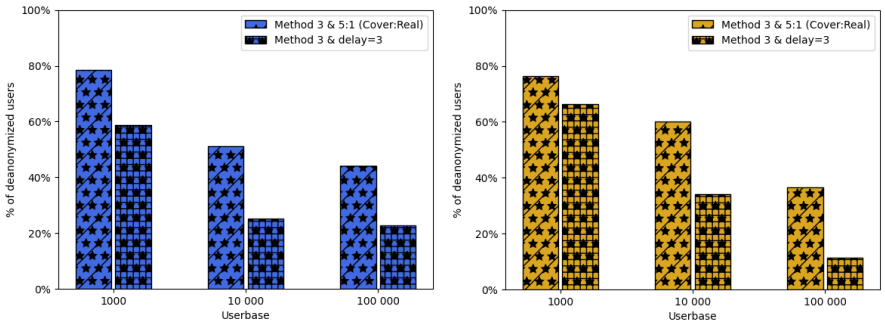


Fig. 9. The impact of delay and cover traffic on the effectiveness of various methods in user-topic linking (sending rate group is $(100 - 1{,}000)$, round length = 1 hour). (a) Twitter ($a = 5$, $b = 1$), (b) Reddit ($a = 7$, $b = 1$).

Fig. 9(a) and Fig. 9(b) show the effect of delaying messages and generating cover messages on the percentage of de-anonymized users when method 3 is considered. The delay, like in the first attack, is shown to be a better countermeasure than cover traffic. For example, if the userbase size is 10,000 and the maximum delay is 3 hours, the percentage of the de-anonymized users is reduced from 57 % to 25 %. While using five cover messages for each real message (i.e., 5:1) can only decrease the percentage of de-anonymized users to 51 %, see Fig. 9(a).

Overall, the user-pseudonym linking attack is far more effective than the user-topic linking attack. That is mainly for two reasons. First, it is due to the many-to-many relationship between users and topics, whereas users and pseudonyms have a one-to-one relationship. The user may publish on various topics every round, and each topic can get messages from different users over rounds. The second reason is that some of the topics can be *active* for the majority of the

time. Thus, it is difficult to distinguish whether the user publishes on that topic or on another one.

5 Related Work

Anonymous microblogging. Several systems have been proposed to support anonymous microblogging scenarios. The methods used to achieve sender anonymity differ between these systems. The commonly used methods are *mixnets* (Atom [16] and Riffle [15]), *DCnets* (Dissent [5]), *private information retrieval* (Blinder [1], Riposte [4], 2PPS [10], and Spectrum [20]), and *random forwarding* (AnonPubSub [8]). For receiver anonymity, systems like Dissent, Riposte, Blinder, Atom, and Spectrum depend on the concept of broadcasting messages to all users. While systems like AnonPubSub, 2PPS, and Riffle have addressed this goal by proposing anonymous multicast communication mechanisms.

Intersection Attacks. Many studies on intersection attacks and statistical disclosure attacks have been conducted. In [7, 14, 25], researchers demonstrated the efficacy of statistical disclosure attacks against mixnets-based systems, especially when the systems support full bidirectional communications [7]. Statistical disclosure attacks have been proven to be effective in attacking the Signal application's sealed sender mechanism in order to deduce the relationship between the sender and the recipient of an end-to-end encrypted message stream [18]. Intersection attacks combined with social network analysis were shown in [26] to be capable of determining the social relationships of a targeted social network user. A variant of statistical disclosure attacks based on an Expectation-Maximization algorithm has also been demonstrated to be feasible on anonymous email networks [23].

6 Conclusion & Future Work

In this paper, we conducted intersection attacks in anonymous microblogging against pseudonym-based and topic-based messaging patterns. The findings demonstrate that the attacks are effective and practical in de-anonymizing users, particularly when they post messages under pseudonyms. In the user-topic linking attack, increasing the user base has proven to be a far more effective mitigation solution than in the user-pseudonym linking attack. The users with high sending rates are found to be more vulnerable to the attacks, especially when the number of these users is small. We evaluated the impact of using delay and cover traffic, which are common mitigation techniques against intersection attacks. According to our results, delaying messages for several hours can reduce the performance of intersection attacks better than cover traffic. However, both delay and cover traffic do not completely prevent intersection attacks because they only extend the time at which users lose their anonymity. Furthermore, they introduce significant latency and bandwidth overhead, making them less convenient for all scenarios.

In the future, we would like to investigate even more improved and sophisticated intersection attacks in the topic-based messaging pattern. Additionally, we will work on developing suitable mitigation approaches which can create stable, long-lived anonymity sets without imposing high latency or bandwidth overhead on users while considering realistic user communication behavior, such as users' ability to join the anonymity system at any time and have various sending rates.

Acknowledgements. This work was partially supported by funding from the German Research Foundation (DFG), research grant 317688284. We thank Tim Grube for his insightful comments on an earlier draft of the manuscript. We would also like to thank the anonymous NordSec reviewers for their feedback.

References

1. Ittai Abraham, Benny Pinkas, and Avishay Yanai. Blinder–scalable, robust anonymous committed broadcast. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 1233–1252, 2020.
2. O. Berthold et al. Dummy traffic against long term intersection attacks. In *International Workshop on Privacy Enhancing Technologies*, pages 110–128. Springer, 2002.
3. Raymond Cheng, William Scott, Elisaweta Masserova, Irene Zhang, Vipul Goyal, Thomas Anderson, Arvind Krishnamurthy, and Bryan Parno. Talek: Private group messaging with hidden access patterns. In *Annual Computer Security Applications Conference*, pages 84–99, 2020.
4. H. Corrigan-Gibbs et al. Riposte: An anonymous messaging system handling millions of users. In *2015 IEEE Symposium on Security and Privacy*, pages 321–338. IEEE, 2015.
5. Henry Corrigan-Gibbs and Bryan Ford. Dissent: accountable anonymous group messaging. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 340–350, 2010.
6. G. Danezis et al. Statistical disclosure or intersection attacks on anonymity systems. In *International Workshop on Information Hiding*, pages 293–308. Springer, 2004.
7. G. Danezis et al. Two-sided statistical disclosure attack. In *International Workshop on Privacy Enhancing Technologies*, pages 30–44. Springer, 2007.
8. J. Daubert et al. Anonpubsub: Anonymous publish-subscribe overlays. *Computer Communications*, 76:42–53, 2016.
9. S. Dixon. Number of twitter users worldwide from 2019 to 2024, 2022. https://www.statista.com/statistics/303681/twitter-users-worldwide/ (Accessed 26-July-2022).
10. S. A. Gaballah et al. 2pps—publish/subscribe with provable privacy. In *2021 40th International Symposium on Reliable Distributed Systems (SRDS)*, pages 198–209. IEEE, 2021.
11. Tim Grube, Markus Thummerer, Jörg Daubert, and Max Mühlhäuser. Cover traffic: A trade of anonymity and efficiency. In *International Workshop on Security and Trust Management*, pages 213–223. Springer, 2017.
12. Jamie Hayes, Carmela Troncoso, and George Danezis. Tasp: Towards anonymity sets that persist. In *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society*, pages 177–180, 2016.

13. M. Karissa et al. Truthy: Enabling the Study of Online Social Networks. In *Proc. 16th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion (CSCW)*, 2013.
14. D. Kedogan et al. Limits of anonymity in open environments. In *International Workshop on Information Hiding*, pages 53–69. Springer, 2002.
15. A. Kwon et al. Riffle: An efficient communication system with strong anonymity. *Proc. Priv. Enhancing Technol.*, 2016(2):115–134, 2016.
16. A. Kwon et al. Atom: Horizontally scaling strong anonymity. In *Proceedings of the 26th Symposium on Operating Systems Principles*, pages 406–422, 2017.
17. Shaahin Madani. *Improving security and efficiency of mix-based anonymous communication systems*. PhD thesis, RMIT University, 2015.
18. I. Martiny et al. Improving signal's sealed sender. *NDSS. The Internet Society*, 2021.
19. K. McKelvey et al. Design and prototyping of a social media observatory. In *Proceedings of the 22nd international conference on World Wide Web companion*, WWW '13 Companion, pages 1351–1358, 2013.
20. Z. Newman et al. Spectrum: High-bandwidth anonymous broadcast with malicious security. *Cryptology ePrint Archive*, 2021.
21. Andreas Pfitzmann and Marit Hansen. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management, 2010.
22. Ania M Piotrowska. Studying the anonymity trilemma with a discrete-event mix network simulator. In *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*, pages 39–44, 2021.
23. J. Portela et al. Disclosing user relationships in email networks. *The Journal of Supercomputing*, 72(10):3787–3800, 2016.
24. Catherine Thorbecke. Facebook says government requests for user data have reached all-time high, 2019. https://abcnews.go.com/Business/facebook-government-requests-user-data-reached-time-high/story?id=66981424 (Accessed 26-July-2022).
25. C. Troncoso et al. Perfect matching disclosure attacks. In *International Symposium on Privacy Enhancing Technologies Symposium*, pages 2–23. Springer, 2008.
26. Alejandra Guadalupe Silva Trujillo, Ana Lucila Sandoval Orozco, Luis Javier García Villalba, and Tai-Hoon Kim. A traffic analysis attack to compute social network measures. *Multimedia Tools and Applications*, 78(21):29731–29745, 2019.
27. D. Wolinsky et al. Hang with your buddies to resist intersection attacks. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 1153–1166, 2013.
28. Tianyin Xu, Yang Chen, Xiaoming Fu, and Pan Hui. Twittering by cuckoo: decentralized and socio-aware online microblogging services. In *Proceedings of the ACM SIGCOMM 2010 Conference*, pages 473–474, 2010.

P4

MITIGATING INTERSECTION ATTACKS IN ANONYMOUS MICROBLOGGING

This chapter was first published as

Sarah Abdelwahab Gaballah, Thanh Hoang Long Nguyen, Lamya Abdullah, Ephraim Zimmer, and Max Mühlhäuser. "Mitigating Intersection Attacks in Anonymous Microblogging." In 18th International Conference on Availability, Reliability and Security (ARES), pp. 1-11. ACM, 2023.

Contribution Statement: I led the idea generation, conceptualization, and development of the proposed work, as well as the experiment design, data analysis, and writing of the publication. All co-authors helped with critiques and comments on the concept design and participated in the creation of the publication.

Mitigating Intersection Attacks in Anonymous Microblogging

Sarah Abdelwahab Gaballah
Telecooperation Lab (TK), Technical University of Darmstadt
Darmstadt, Germany
gaballah@tk.tu-darmstadt.de

Thanh Hoang Long Nguyen
Telecooperation Lab (TK), Technical University of Darmstadt
Darmstadt, Germany
long.nguyen@stud.tu-darmstadt.de

Lamya Abdullah
Telecooperation Lab (TK), Technical University of Darmstadt
Darmstadt, Germany
abdullah@tk.tu-darmstadt.de

Ephraim Zimmer
Telecooperation Lab (TK), Technical University of Darmstadt
Darmstadt, Germany
zimmer@privacy-trust.tu-darmstadt.de

Max Mühlhäuser
Telecooperation Lab (TK), Technical University of Darmstadt
Darmstadt, Germany
max@tk.tu-darmstadt.de

ABSTRACT

Anonymous microblogging systems are known to be vulnerable to intersection attacks due to network churn. An adversary that monitors all communications can leverage the churn to learn who is publishing what with increasing confidence over time. In this paper, we propose a protocol for mitigating intersection attacks in anonymous microblogging systems by grouping users into anonymity sets based on similarities in their publishing behavior. The protocol provides a configurable communication schedule for users in each set to manage the inevitable trade-off between latency and bandwidth overhead. In our evaluation, we use real-world datasets from two popular microblogging platforms, Twitter and Reddit, to simulate user publishing behavior. The results demonstrate that the protocol can protect users against intersection attacks at low bandwidth overhead when the users adhere to communication schedules. In addition, the protocol can sustain a slow degradation in the size of the anonymity set over time under various churn rates.

CCS CONCEPTS

• **Security and privacy** → Network security; Anonymity; • **Social Networks** → Microblogging.

KEYWORDS

Anonymous Communication, Anonymous Microblogging, User Publishing Behavior, Traffic Analysis, Intersection Attacks, Mitigation, Communication Schedules

ARES 2023, August 29–September 01, 2023, Benevento, Italy

ACM ISBN 979-8-4007-0772-8/23/08…$15.00
https://doi.org/10.1145/3600160.3600166

ACM Reference Format:
Sarah Abdelwahab Gaballah, Thanh Hoang Long Nguyen, Lamya Abdullah, Ephraim Zimmer, and Max Mühlhäuser. 2023. Mitigating Intersection Attacks in Anonymous Microblogging. In *The 18th International Conference on Availability, Reliability and Security (ARES 2023), August 29–September 01, 2023, Benevento, Italy.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3600160.3600166

1 INTRODUCTION

Microblogging is a popular form of online social networking that enables the rapid dissemination of information and news. Platforms that support microblogging, such as Facebook and Twitter, have played a substantial role during sociopolitical protests and crisis situations, such as the 2022 Iran protests [31] or the 2023 Turkey-Syria earthquake [10]. However, freely expressing one's views on these platforms may have serious ramifications. An activist who is caught posting about a regime-critical topic, for example, may face serious legal consequences [30]. Additionally, by observing which topics a user is publishing on, service providers can deduce sensitive information such as health issues, financial status, or sexual preferences. Creating fake accounts on these platforms is a popular strategy to hide real identities. However, this does not solve the problem because communication metadata, such as the user's IP address, can be used by the platforms to associate the fake account with the user's location or identity.

Over the last years, many anonymous communication systems have been proposed to protect communication metadata, with some of these systems mainly designed for microblogging [1, 4, 12, 20, 21]. An anonymous communication system can initially ensure that a user cannot be identified among a group of other users, known as an anonymity set [27]. However, the anonymity sets change over time due to network churn. This change in the anonymity sets makes users susceptible to traffic analysis attacks.

Intersection attacks are one of the strongest traffic analysis attacks [2, 7, 8, 19, 23, 29, 33]. These attacks could be applied against almost any existing practical anonymous communication system [17]. In these attacks, an adversary who monitors the communication can intersect the anonymity sets over time to single out a certain user [2]. An example of these attacks is when a corrupt mayor discovers that someone in the city has created a Facebook account with a fictitious name and exposes information about the mayor's

corruption or illegal/immoral act. To determine who owns this account, the mayor forces the internet service provider of the city to provide him with a list of the names of people who are using Facebook (or connecting to the Internet) whenever a new post is published on the targeted account. Each list may contain many users; however, when the mayor intersects these lists, the size of the resulting set decreases over time until it contains only one user, the account's owner. These attacks are applicable even if the account's owner connects to Facebook using an anonymous communication system.

Intersection attacks are very powerful, particularly when launched for a long time. Their mode of operation typically falls into the category of passive attacks, which means users will not become aware of the fact that an attack is taking place. They can either be performed deterministically, meaning that, in case the attack is successful, the adversary is able to link a user to her fake account with absolute certainty. Or they can be performed probabilistically—this variant is called a statistical disclosure attack—which aims at estimating the likelihood that a target user was the owner of a specific account among a group of users [17].

Many solutions have been introduced in the literature to mitigate the intersection attacks. These solutions include sending dummy/cover messages (also known as cover traffic) to hide the real communication [9, 22] or delaying the users' messages on the anonymous communication system side for a random amount of time [20]. Supporting a wide userbase was also recommended in [4] as a strategy to increase anonymity sets and thereby hinder intersection attacks. Nevertheless, all these solutions have been shown to be ineffective [2, 11, 16]. Anonymous communication systems, such as [1, 12, 21], consider constant user participation, i.e., the requirement for users to always be online and send messages to the system, as the only way to effectively protect against intersection attacks. However, this requirement is not realistic or practical. A framework for vulnerability monitoring and active mitigation of anonymity loss under intersection attacks was proposed in [33]. Nevertheless, this framework incurs considerable bandwidth and latency costs due to the inefficient method it uses to build anonymity sets and the random assignment of users to fixed sets of the same size.

In this paper, we propose a protocol for protecting users who publish messages on an anonymous microblogging system from being de-anonymized (i.e., linked to their published content) by intersection attacks. For the sake of efficiency, the protocol works by forming anonymity sets based on the similarity of the users' publishing behavior. It creates a communication schedule for users in each anonymity set to control message transmissions in such a way that users within a set behave indistinguishably from the point of view of an adversary. The schedules can be adjusted to optimize the trade-off between bandwidth overhead and latency based on the users' performance needs. Our protocol focuses only on protecting users when they are publishers on an anonymous microblogging system, so it is out of our scope to protect them when they are subscribers.[1]

[1]In this case, users can be protected by broadcasting published messages, i.e., the system sends every published message to all users, as seen in [1, 4, 5]. However, broadcasting imposes a significant communication overhead on users, making it an inefficient solution. As a result, more research in this area is still clearly required.

The paper's main contributions are: (1) a protocol that prevents intersection attacks by grouping users into sets according to how they publish and establishing communication schedules that enforce indistinguishability across users in the same set; (2) an analysis of realistic user behavior with the help of real-world datasets from Twitter and Reddit; and (3) an evaluation of the protocol, in which we study the impact of the schedule design on bandwidth and latency, as well as the impact of the churn rate on the size of the sets provided by our protocol.

This paper is organized as follows: Section 2 introduces the system and threat models. Section 3 then presents our protocol and its five phases. Following that, in Section 4, we discuss the evaluation results of our experiments. Section 5 presents a discussion on some additional settings in our protocol. In Section 6, we provide a review of the related work. Finally, Section 7 concludes the paper and presents future work.

2 MODELS

This section describes the system model and the design assumptions of our mitigation protocol. It also discusses the adversary's capabilities and goals.

2.1 System Model

Our mitigation protocol is assumed to be employed by an anonymous microblogging system. We do not restrict the system to any particular anonymity technique. Instead, we consider a broadly applicable decentralized system based on an anytrust model, i.e., a system run by many servers (e.g., many mix nodes) where at least one of them is trustworthy [21]. Even if some of the system's servers are malicious, the system is assumed to be honest in its execution of the protocol.

The system allows users to publish posts under pseudonyms on a shared board (e.g., a public bulletin board). However, the users are able to publish their posts only when the protocol permits. The protocol is carried out for every set of new users U (we call U a "batch"), where the size of U should be above a pre-defined and system specific value. Each user $u_i \in U = \{u_1, u_2, ...u_n\}$ has only one pseudonym $p_j \in P = \{p_1, p_2, ...p_n\}$ and vice versa. The system is responsible for ensuring *unlinkability* between u_i and p_j.

The communication in the system is assumed to proceed in time intervals $T = \{T_1, T_2, ...T_v\}$. Each time interval $T_e \in T$ consists of a set of time slots $\{t_1, t_2, ..., t_w\}$, where $|T_e| = w$. The users send their messages during the slots. Users can send at any time during the slot period, but the content of the messages will be published publicly by the system only at the end of the slot.

All the exchanged messages between the users and the system should be encrypted, and padded to the same length. To prevent an adversary from probabilistically profiling a user based on rates of sending, every user that wants to send in a time slot $t_l \in T_e$ must send m messages. To reach the required number of messages m, the user can send cover messages if the number of her actual messages is less than m. If a user has more than m real messages in t_l, the extra messages should be delayed in a *message queue* on the user's side where they can be sent later during the next slot(s).

When users send cover messages, these messages will not be part of the published content, as they are just used to feign identical communication behavior among the users. Since the content of users' real messages will be published publicly and can be read by anyone, users must not include any personally identifiable information, such as real names or addresses, in the content.

2.2 Threat Model

We assume a global passive adversary $\mathcal{A}$ who observes the whole communication. $\mathcal{A}$ can only see the message's metadata but not the content. It does not alter, delay, or drop packets sent by users. Also, we do not consider the ability of $\mathcal{A}$ to launch Sybil attacks. Additionally, we assume that $\mathcal{A}$ cannot corrupt the functionality of the system or deny its availability. Moreover, it cannot control the whole system, thus it cannot break the unlinkability property that is provided by the system to link a user to her pseudonym. To de-anonymize users, $\mathcal{A}$ utilizes an intersection attack. It launches the attack by observing and analyzing the publishing behavior of the users. When there is no change in the behavior of users belonging to the same set, $\mathcal{A}$ fails to de-anonymize users, and hence intersection attacks are rendered ineffective. Any de-anonymization attack based on analyzing the published content—e.g., analyzing the writing styles—is out of our scope.

3 PROTOCOL ARCHITECTURE

In this section, we describe our mitigation protocol in detail. The protocol is divided into five phases: the *Arrival Phase* (Section 3.1), the *Learning Phase* (Section 3.2), the *Grouping Phase* (Section 3.3), the *Scheduling Phase* (Section 3.4), and the *Communication Phase* (Section 3.5). The protocol is carried out in batches, with each batch of new users going through its own set of phases. Figure 1 illustrates a general process diagram of the protocol's phases.

3.1 Arrival Phase

During this phase, the protocol waits until it collects a batch of new users U who want to join the microblogging system. When the size of the batch U reaches the pre-defined batch threshold, the users in the batch are notified to begin the *Learning Phase*, during which they are able to communicate and publish their messages. If a user $u_i \in U$ has a message to publish, but the size of U is still less than the threshold, the message should be delayed in the user's message queue and sent only to the system when u_i enters the *Learning Phase*. In Section 5, we go into further depth about the delay that is introduced during the *Arrival Phase*.

After a complete batch enters the *Learning Phase*, the protocol can start a new *Arrival Phase* to accumulate a new batch.

3.2 Learning Phase

In this phase, the protocol has a batch U, and needs to learn the publishing behavior of the users in U during a time interval $T_{learning}$ (where $T_{learning} = T_1$), i.e., in which time slots the users send real messages to be published. Since the system that executes the protocol is not totally trusted (we assume an anytrust model, cf. Section 2.1), the protocol is not able to learn the publishing behavior of the users directly. Instead, it learns the publishing behavior on the pseudonyms in P, i.e., in which time slots the pseudonyms have messages. This learned behavior is used to reflect the behavior of the users, who are the owners of the pseudonyms.

We refer to the publishing behavior on a pseudonym $p_j \in P$ as a binary vector $B_{p_j} \in \{0,1\}^w$. To learn the publishing behavior on pseudonyms, the protocol requires every $u_i \in U$ to send m messages (real or cover) in every $t_l \in T_{learning}$. Each real message sent by the user should specify the actual slot when the message was created on the user's side. The message's creation slot may differ from the message's sending slot. That happens because the messages might be delayed on the user's side until the *Learning Phase* starts or because the user has more messages than what is allowed to be transmitted in a slot.

The protocol computes B_{p_j} in every t_l as follows:

$$B_{p_j,t_l} = \begin{cases} 1 & \text{if } p_j \text{ has real messages created at } t_l \\ 0 & \text{if } p_j \text{ has no real messages created at } t_l \end{cases} \quad (1)$$

By the end of the *Learning Phase*, the protocol has a binary vector for each pseudonym $p_j \in P$ which demonstrates the publishing behavior on this pseudonym during $T_{learning}$. In this vector, when a slot has a value of 1, it means the p_j's owner created real message(s) during this slot, while a value of 0 indicates the opposite.

3.3 Grouping Phase

The goal, in this phase, is to group the pseudonyms in the set P based on the similarity in their publishing behavior. To achieve this goal, a k-mode clustering algorithm is employed [18]. We consider the k-mode algorithm due to its capability of grouping data points efficiently in such a way that minimizes the total mismatches between the corresponding attribute values of the two data points. The protocol performs clustering by carrying out the following steps.

- Randomly select a set of k pseudonyms from P. [2] The publishing behavior vectors of the chosen pseudonyms are the initial cluster heads, with one vector assigned to each head. The set of clusters is defined as $S = S_1, S_2, ..., S_k$, and the head of a cluster $S_x \in S$ is denoted as $S_x^{head} \in \{0,1\}^w$.
- Calculate the distance between every head S_x^{head} and the publishing behavior vector of every pseudonym $p_j \in P$. The distance between the two vectors, B_{p_j} and S_x^{head}, is computed as the total mismatches in each slot's value in the two vectors. The smaller the number of mismatches is, the less the distance is (i.e., the more similar the two vectors are). This distance measure is known as the Simple Matching Coefficient (SMC) [18]. Formally, the distance is calculated as follows:

$$dist(B_{p_j}, S_x^{head}) = \sum_{l=1}^{w} \delta(B_{p_j,t_l}, S_{x,t_l}^{head}) \quad (2)$$

where

$$\delta(B_{p_j,t_l}, S_{x,t_l}^{head}) = \begin{cases} 0 & \text{if } B_{p_j,t_l} = S_{x,t_l}^{head} \\ 1 & \text{if } B_{p_j,t_l} \neq S_{x,t_l}^{head} \end{cases} \quad (3)$$

- Assign each pseudonym p_j to a cluster S_x whose head S_x^{head} is the most similar to the behavior vector B_{p_j}.

[2] There are some methods to determine the best value of k such as the elbow method [13].

Figure 1: A process diagram of the protocol's phases

- Update S_x^{head} for each cluster $S_x \in S$. The new instance of S_x^{head} is calculated as the mode of the behavior vectors of all pseudonyms in S_x. Thus, $S_{x,t_l}^{head} = 1$ if t_l in most of the vectors has a value of 1; otherwise, S_{x,t_l}^{head} is assigned a value of 0.
- Repeat steps 2-4 until there are no more changes in the clusters, i.e., no updates in the cluster heads and/or the members of the clusters.

The output of the previous steps is a set of clusters S, where each cluster $S_x \in S$ consists of a set of pseudonyms that are similar in terms of publishing behavior vectors.

3.4 Scheduling Phase

To prevent $\mathcal{A}$ from de-anonymizing users using intersection attacks, all owners of pseudonyms belonging to a set S_x must communicate in an indistinguishable manner. To achieve that, the protocol creates a schedule $H_x \in \{0,1\}^w$, that specifies the time slots on which the owners of the pseudonyms in S_x should send their messages during the time intervals of the *Communication Phase*. In our protocol, we construct the communication schedules for the users to communicate in the future based on their communication history. Initially, in this phase, the schedule H_x is computed based on the publishing behavior vectors learned during the *Learning Phase* (Section 3.2). Later, in the *Communication Phase* (Section 3.5), we discuss how the schedules can be updated.

This phase consists of two steps:

(1) *Creation*: H_x consists of w time slots, where the value of each slot $t_l \in H_x$ is computed as:

$$H_{x,t_l} = \begin{cases} 1 & \text{if } \sum_{j=1}^{|S_x|} B_{p_j,t_l} \geqslant q \\ 0 & \text{if } \sum_{j=1}^{|S_x|} B_{p_j,t_l} < q \end{cases} \quad (4)$$

where q refers to the activity threshold in each t_l.
Each time slot in the schedule H_{x,t_l} is assigned a value of either 1 or 0 based on the total number of publishing behavior vectors having t_l with a value of 1 (i.e., activity). During the *Communication Phase*, each user who owns a pseudonym $p_j \in S_x$ must send a message during a slot t_l when the value of $H_{x,t_l} = 1$ (we refer to t_l in this case as an *active slot*). While $H_{x,t_l} = 0$ means that the user must not send any message during that slot, because t_l in this case is *inactive slot*. An example of how to create a schedule is shown in Figure 2.

(2) *Broadcasting*: After the schedules are created, all users receive information about the clustered pseudonyms and their schedules (obviously, pseudonyms that belong to the same cluster have the same schedule). Each user identifies the corresponding cluster of her pseudonym and the related schedule. Since the protocol does not recognize the pseudonyms of the users, it cannot directly send each user her designated schedule.[3]

The bandwidth overhead and latency introduced by applying a schedule H_x are highly influenced by the value of q. When it is low, many slots in H_x may be assigned a value of 1. Consequently, the users will be required to send messages in many time slots during the *Communication Phase*. Thus, the low value of q may result in a high bandwidth overhead for the users during the *Communication Phase*. When the q value is high, i.e. the opposite case, it may not be easy to reach the threshold. Therefore, H_x could contain only a low number of *active slots*, probably resulting in a high latency for many users of the corresponding cluster during the *Communication Phase*, especially for users with high publishing rates. In the evaluation section, we discuss the impact of q on the bandwidth overhead and latency in greater detail.

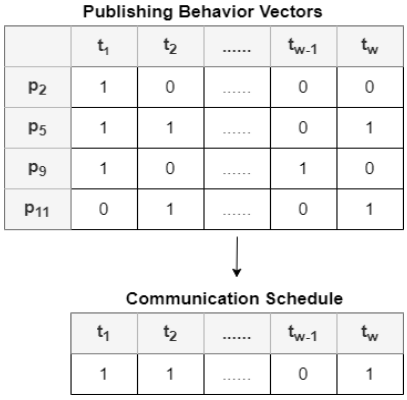


Figure 2: Example for the scheduling where the activity threshold is 50%.

3.5 Communication Phase

By the end of the *Scheduling Phase*, every user $u_i \in U$ is assigned a schedule H_x, where u_i's pseudonym $p_j \in S_x$. The user u_i must strictly follow the schedule, which means that u_i should send messages when $H_{x,t_l} = 1$ and refrain from sending any message when $H_{x,t_l} = 0$. The number of messages that u_i must send when $H_{x,t_l} = 1$ is m. If u_i is scheduled to send in a specific time slot but does not have real messages to send, she should send cover messages. When u_i creates real messages during t_l, and according to the schedule, she must not send in that time slot, the created messages shall be

[3]The amount of overhead introduced by this broadcasting step depends on the batch size. The overhead can be avoided if the system allows anonymous retrieval, e.g., by using a private information retrieval technique [12, 21]. Each user can then retrieve only the information pertinent to her schedule without the system being able to tell what the retrieved information is.

pushed to a message queue that resides on the user's side.

When the users strictly follow the schedules that are assigned to them, their anonymity sets will be *indistinguishability sets*.

Definition. *S' is an indistinguishability set if all users in this set have the same behavior when they send messages to the system. The probability for $\mathcal{A}$ guessing the pseudonym $p_j \in S_x$ of a user $u_i \in S'$ is $1/|S'|$.*

Intuitively, this means that $\mathcal{A}$ can de-anonymize u_i (link a pseudonym to u_i) by only making random guesses. If one user in S' slightly deviates from the behavior of the other users in the set, the protocol will not be able to guarantee indistinguishability for this user. The larger the size of the indistinguishability set $|S'|$ is, the more protected the users are. To guarantee a minimum level of indistinguishability, the protocol must ensure that the size of every indistinguishability set $|S'|$ is larger than a certain number z.

Churn. A churn in the indistinguishability sets occurs when users do not send in an *active slot* in their schedules. In our protocol, we assume that the churn occurs only due to unintentional reasons, e.g., the users fail to send due to a network connection problem. Hence, we do not take into account when users fail to adhere to the schedule due to active attacks, such as delaying or dropping messages by an adversary (cf. Section 2.2).

Elimination. When a user u_i does not follow the schedule in one *active slot*, the protocol supports two settings:

- *No chances:* u_i will be removed from her set and no longer be able to publish under her pseudonym. The user is eliminated because she has deviated from the behavior that is obliged by the schedule, i.e., she behaved differently compared to the other users in her set. As a result, the protocol can no longer guarantee indistinguishability for her. If u_i wants to publish posts on the system again, she must join the system as a new user with a new identity. That means she will be part of a new batch and go through the mitigation protocol phases again.
- *Chances:* It may not be practical to eliminate u_i if she does not follow the schedule in a single *active slot*. Thus, in order to maintain a level of practicality without breaking the indistinguishability, the protocol imposes a delay time d allowing to wait for u_i to send her messages. Therefore, when u_i does not send messages in an *active slot*, the messages sent by other users who belong to the same set of u_i will be delayed. It will be published on the system either when u_i sends her messages before the d period ends or when the waiting time exceeds d. In the latter case, u_i will be eliminated. In the beginning of the *Communication Phase*, the protocol can assign each user a number of failure times, i.e., a user is allowed to fail to follow the schedule up to this number.

Updating the schedule. In the previous phase, a schedule H_x is created based on the publishing behavior on pseudonyms during $T_{learning}$. In the *Communication Phase*, the protocol can update H_x to adjust it to the recent history of the publishing behavior. That means H_x in T_{e+1} will be based on the publishing behavior during T_e. To accomplish this, the steps below are carried out for each time interval T_e:

- Compute a new instance of B_{p_j} for each pseudonym p_j during the time interval T_e using the equation 1.
- Create H_x^{Temp} using the equation 4.
- Update H_x to equal H_x^{Temp}, if H_x^{Temp} contains at least a certain number of *active slots*.

4 EVALUATION

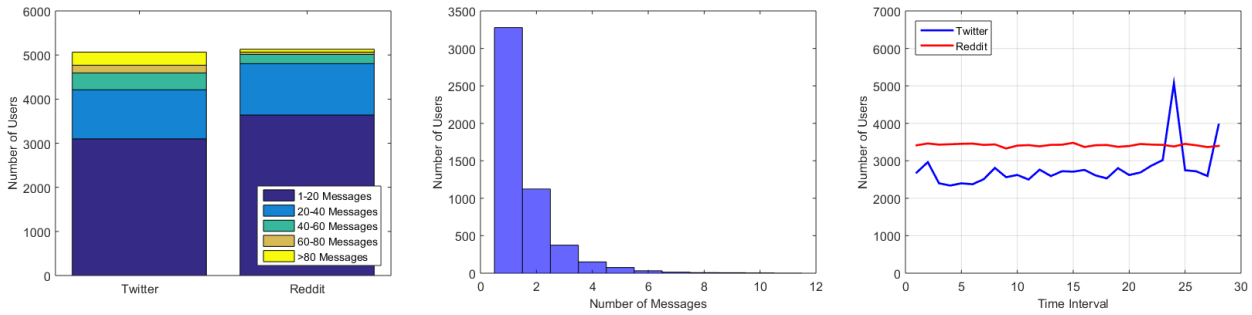
In this section, we analyze realistic user publishing behavior in microblogging settings and assess the efficiency of our proposed mitigation protocol in light of this behavior. A prototype of our protocol is implemented in Python. The user publishing behavior in the prototype is derived from two real-world datasets collected from two popular microblogging platforms, Twitter and Reddit. The batch threshold is set to 5000. The number of slots w in a time interval T_e is 24, and the size of each slot $t_l \in T_e$ is 1 hour (in accordance with related work [11, 17]). Therefore, the *Learning Phase* in our experiments lasts for 24 hours. The message size is assumed to be 1 KB as the messages in microblogging scenarios are typically small, e.g., the text content of a Tweet can contain up to 280 characters or Unicode glyphs [28]. The number of messages m that a user can send in a time slot is set to 1 (as in [4, 12, 17, 21]). The number of the clusters/sets k is 15 which is chosen using the elbow method. The minimum size z of every set S_x is set to 50.

4.1 Datasets

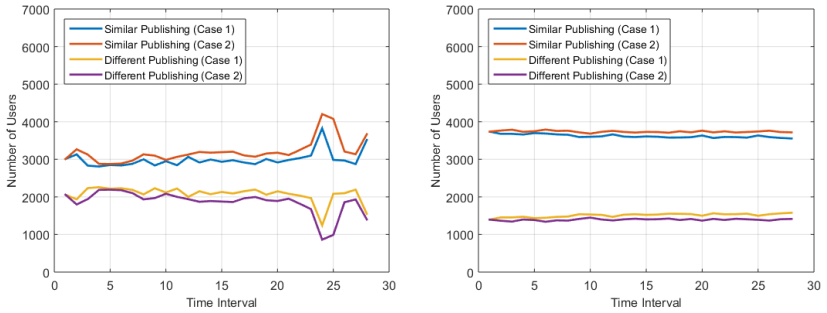
We used two datasets in our evaluation. The first dataset is an already existing collection of records extracted from Twitter over the course of the entire month of November 2012 [24][25]. This dataset contains 22534846 tweets, 6914561 users, and 3379976 topics, referred to as hashtags. The second dataset is collected by us from Reddit for the whole month of October 2021. This dataset contains posts and comments from 1638157 different users and 3403 different topics, referred to as subreddits. Both datasets include a timestamp, a user id, and a topic (hashtag/subreddit) at each record.

4.2 Analysis of Realistic User Publishing Behavior

User publishing behavior has a considerable impact on the various phases of the protocol. Thus, we examined user behavior in batches from the two datasets. Figure 3a displays the total publishing rates of users over the course of a month, where the batch threshold is 5000. As shown in the figure, the majority of Twitter and Reddit users sent between 1 and 20 messages during the month. Twitter users publish at a higher rate than Reddit users; for example, the number of users who send more than 60 messages on Twitter is much higher than on Reddit. The difference in publishing rates between Twitter and Reddit users is to be expected, given the two platforms' different service models. The distribution shown in Figure 3a was found to be nearly the same for various batch threshold values (we ran the analysis for values of 1000, 2000, 3000, 5000, and 7000).



(a) The distribution of users based on total publishing rates. **(b) Twitter: Histogram of the publishing rates during the *Learning Phase*.** **(c) The number of users who do not publish during each Time Interval.**



(d) Twitter: The change in user publishing behavior over intervals. **(e) Reddit: The change in user publishing behavior over intervals.**

Figure 3: Analysis of user publishing behavior

Figure 3b shows the low publishing rates of Twitter users during the *Learning Phase*. That means the vectors created during this phase typically have a limited number of slots observed with real messages. The low publishing rates during the *Learning Phase* were also noticed among Reddit users. Additionally, Figure 3c illustrates that a large number of users do not publish during each time interval. That, again, emphasizes the low publishing rates of users.

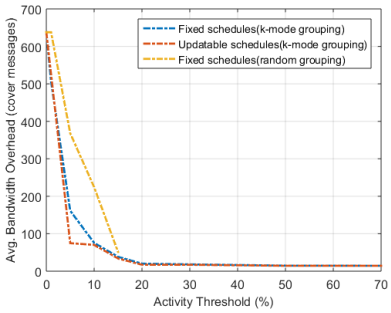
We also looked at how the user publishing behavior evolves over time. This investigation is necessary to understand whether a single fixed schedule for each set is sufficient or whether the schedule should be updated regularly. To accomplish this, we considered two cases:

- Case 1: The user's publishing behavior during the first interval T_1 (which represents the *Learning Phase* in our protocol) is compared to her behavior during the subsequent intervals $T_2, T_3, ...T_v$. If they are similar, then a schedule created based on her behavior in the *Learning Phase* will be appropriate for the communication during T_e. However, if they are different, the schedule will be less suitable for communication during T_e.
- Case 2: The user's publishing behavior during every two consecutive intervals T_e and T_{e+1} is compared. If they are similar, a schedule created based on her behavior during interval T_e is suitable for communication during the interval T_{e+1}.

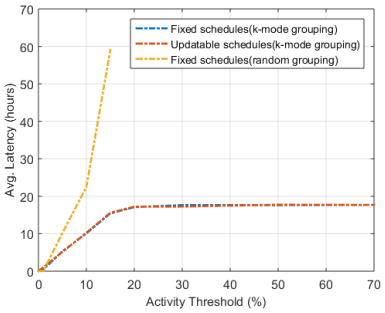
The simple matching coefficient (SMC) is used in both cases to determine the similarity and dissimilarity of the publishing behavior. Figures 3d and 3e show the number of users who exhibit similar or different behavior in each time interval based on the aforementioned cases; the results are illustrated for Twitter and Reddit batches, respectively. For instance, in Figure 3d, during time interval number 5, there are roughly 2900 users with publishing behavior similar to their publishing behavior in the first interval (the *Learning Phase*) based on Case 1. Two publishing behavior vectors were deemed similar in our analysis if they were identical or only differed in one slot.

The two figures depict that, in both Case 1 and Case 2, there are more users with similar publishing behavior than those with different behavior. Hence, it is a worthwhile endeavor to create schedules based on the users' communication history. Case 2 demonstrates results for similarity higher than Case 1. Although the gap between the results of the two cases is not significant, it still does imply that updating the schedules for intervals might result in more representative schedules and, thus, less bandwidth and latency overhead.

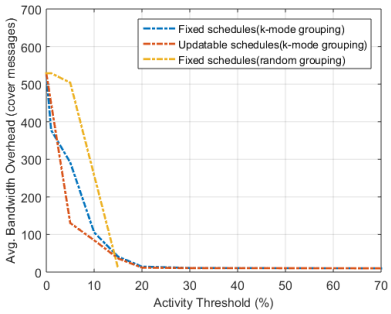
We think that the reason for the dynamic nature of the results in Figure 3d versus those in Figure 3e is that the publishing behavior of Twitter users is more triggered by hot topics and trends, i.e., users tend to publish more or less depending on the presence of hot topics. That does not appear to be the case on Reddit, where users seem to be more consistent in their publishing behavior.



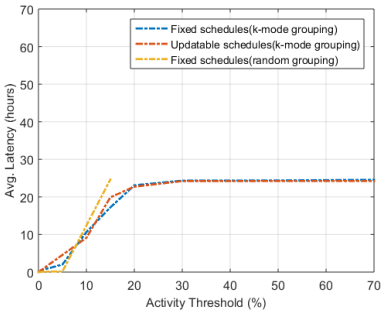
(a) Twitter: The avg. bandwidth overhead per user.



(b) Twitter: The avg. message latency.



(c) Reddit: The avg. bandwidth overhead per user.



(d) Reddit: The avg. message latency.

Figure 4: The impact of activity threshold on bandwidth overhead and latency

4.3 Bandwidth and Latency Overhead

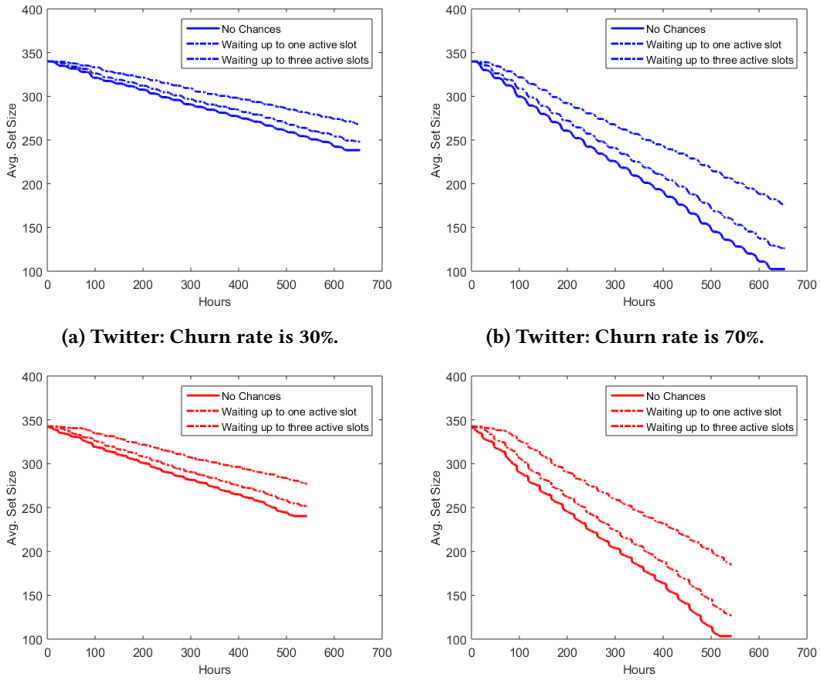
Our protocol aims at protecting users against intersection attacks by requiring users from the same set to communicate indistinguishably by following a schedule. However, communication based on schedules can be costly in terms of bandwidth and latency overhead. Since the activity threshold q, defined in Section 3.4, affects the amount of the overhead, we tested several values for this parameter to assess the efficiency of the schedules. In Figure 4, for instance, when the threshold value is 10%, it means that a time slot t_l is an *active slot* (i.e., $H_{x,t_l} = 1$) if at least there are 10% of the publishing vectors having a value of 1 at t_l.

The bandwidth overhead and latency that the schedules impose are influenced by how the users are grouped. Therefore, we compared how effective the schedules are when the sets are formed using the k-mode algorithm and when they are created randomly (i.e., users are randomly divided into sets during the grouping phase). The random grouping was carried out ten times, and the bandwidth and latency overhead results of each value of q were then averaged. We found that the value of q must be 15% or lower in order to generate schedules for random groups. Even when it is 15%, for some groups (usually 5 to 8 out of the 15 groups), the schedules cannot be created. The behavior vectors in each group are so different from one another, so it is challenging to discover overlapping *active slots* between the vectors. That makes it impossible for the protocol to produce schedules when the value of q is greater than 15%. Nevertheless, that is not the case when the behavior vectors are grouped into sets based on similarity using the k-mode algorithm.

Additionally, we evaluated the overhead when the schedules are fixed and when they are updated during the *Communication Phase*. In our experiments, the schedule of a set S_x is only updated when the new schedule H_x^{Temp} contains at least two *active slots*. We discovered that updatable schedules are feasible only when the k-mode, not random grouping, is employed for grouping. The substantial disparities between the behavior vectors in each group are to blame once more for this.

Bandwidth overhead. The bandwidth overhead per user is calculated in our evaluation by counting the total number of cover messages sent by a user during the *Communication Phase*. Figures 4a and 4c depict the average total bandwidth overhead per user. The fixed and updatable schedules based on k-mode grouping have notably lower bandwidth overhead than the schedules based on random grouping. That is to be expected because the k-mode algorithm provides sets with more similar publishing behavior than random grouping.

In both Figures 4a and 4c , the results show an inverse relationship between the bandwidth overhead and the threshold q. Hence, increasing the value of q leads to lower overhead, and vice versa. This aligns with intuition because a low threshold implies that if a set contains a few publishing vectors with a value of 1 at t_l,



(a) Twitter: Churn rate is 30%. **(b) Twitter: Churn rate is 70%.**

(c) Reddit: Churn rate is 30%. **(d) Reddit: Churn rate is 70%.**

Figure 5: The impact of churn on the average indistinguishability set size.

the protocol will consider this sufficient to make t_l an *active slot*. Accordingly, the resulting schedule pushes all other users in the set (those with vectors with a value of 0 at t_l) to send cover messages in order to achieve indistinguishability.

As illustrated in the figures, the average bandwidth overhead stabilizes at a value of 20% and higher for fixed and updatable schedules that are based on k-mode. This indicates that there is no discernible change in the schedules at these values. The reason for this is that users' publishing rates are low, both overall and during the *Learning Phase*, indicating sparse slots with a value of 1 in the publishing vectors. That makes it difficult to reach the high threshold value required to consider a slot t_l as an *active slot*. As a result, there is no need to increase the threshold value beyond 20%, as this does not lead to any further optimization. Another interesting observation is that the updatable schedules can introduce less bandwidth overhead than fixed schedules only if q is less than 15. This observation holds true for both the Twitter and Reddit results. This could also be due to the low publishing rates, which means that no further improvements can be made when increasing the threshold regardless of whether or not the schedules are updated. Another reason is that the number of users in the batches who do not send messages at all in each interval is very large. At each interval, more than 2500 Twitter users and around 3500 Reddit users do not send any messages (cf. Figure 3c). As a result, most new schedules typically contain only zero values (i.e., *inactive slots*), and the protocol disregards them because they lack at least two *active slots*. That means changes in the schedules do not often happen, which makes the results of the fixed and updatable schedules not that different.

Latency. Figures 4b and 4d demonstrate the average message latency introduced by the schedules during the *Communication Phase*. The message latency is calculated as the time between generating a message on the user's side and publishing it on the system's side. As shown in the figures, the fixed and updatable schedules that are based on k-mode introduce lower latency than those that are based on random grouping. The latency results of fixed schedules are similar to the updatable schedules. The reasons behind this are similar to those explained in the bandwidth section.

In contrast to the results of the bandwidth overhead evaluation, the latency has a direct relationship with the threshold q. That is, increasing the threshold value increases latency, and vice versa. This is understood because when the threshold value is high, a larger number of publishing vectors with a value of 1 at t_l are required to make t_l an *active slot*; this is also difficult given the low publishing rates of users in the two datasets. As a result, the number of *active slots* may be limited, causing many messages to wait on the user's side for some time before being sent.

When we compare the results between the two datasets, the average message latency is higher on Reddit than on Twitter for larger values of the threshold , as shown in Figures 4b and 4d. That makes sense when looking at the publishing rates since Reddit users have lower publishing rates than Twitter users. Again, from the results of both Twitter and Reddit, we see stability at 20% and higher, so there may be no need to raise the threshold value after 20%.

The bandwidth overhead and latency have an inverse relationship, which means that low latency indicates fast message publication at the expense of high bandwidth overhead, and vice versa. Therefore, the threshold value should be chosen in such a way that users in an indistinguishability set get a good trade-off between bandwidth overhead and latency. A low threshold should be chosen if low latency is critical, whereas a high threshold can be used when low bandwidth overhead is most important. [4]

4.4 Anonymity under Network Churn

Naturally, the network will experience churn as some users might be unable to send during *active slots*. Accordingly, the size of the indistinguishability set will inevitably decrease over time. However, the set size should not degrade quickly. We consider the average indistinguishability set size during the communication phase to assess the impact of churn on anonymity. We simulate churn per set by randomly selecting users from the set to not adhere to their schedule during randomly selected time slots. The number of these selected users from each set is referred to as the churn rate. When the rate is, say 50%, it means that 50% of the users in the set will be chosen at random to ignore the schedule during randomly selected time slots. Figures 5a to 5d depict the average indistinguishability set size when the churn rate per indistinguishability set is 30% and 70%. As expected, the average set size decreases faster as the churn rate per set increases.

We compared the results when the protocol does not give users chances to the results when chances are given. We considered two cases for the chances. The first case is when a user fails to send during an *active slot*, the protocol waits until the next *active slot*. If the user does not send the messages from the previous and new slots during this slot, the user will be removed from the set. In the second case, we increase the waiting time to force the protocol to wait for three consecutive *active slots*; if the user has not sent the required messages by then, she is eliminated. Giving users chances when they miss sending in an *active slot*, as shown in the figures, significantly slows down the degradation in the set size, especially when the churn rate is high or the waiting period is increased. In Figure 5b, for example, when the churn rate is 70% and the waiting time is up to three *active slots*, the average set size drops to around 180 by the end of the simulated *Communication Phase*, which is greater than the set size when no chances are given. Even if no such chances are provided and the churn rate is high, users will be protected by sufficiently large indistinguishability sets. Obviously, if the protocol waits longer for users who miss their schedules to send the messages, the publishing of the users' messages may be delayed, i.e., the latency may increase.

We discovered that the majority of users are typically inactive (i.e., not sending real messages) between 12 a.m. and 6 a.m. Thus, the time slots located during these times are not active in most schedules. The reason for this could be that users are not active during these hours due to sleeping. Since users can only be eliminated from their set during *active slots*, any decrease in the set size occurs only during these slots. Therefore, the size of the set remains constant during *inactive slots*.

[4]The value of q can be determined separately for each set to serve the needs and preferences of the users in that set.

5 DISCUSSION

In this section, we discuss some additional protocol settings.

Latency during the arrival phase. A batch of a particular size is required in our protocol to begin the *Learning Phase*. The joining rate of new users determines the latency for gathering the batch. If the joining rate is high enough, the batch threshold will be met rapidly, resulting in low latency. However, if users join slowly, the latency until the *Learning Phase* starts will be high. Despite this limitation, we believe that waiting for a specific number of new users is better than waiting for a certain period of time to collect new users. The former approach is better because it guarantees a large batch of new users from the start, which aids in creating large anonymity sets for users when we group them based on publishing behavior. Therefore, to avoid waiting for a long time, or even forever, the protocol can wait until the number of new users reaches the batch threshold or wait up to a specific amount of time.

Invalid sets. To ensure a minimum level of indistinguishability, the protocol must ensure that the size of each indistinguishability set $|S'|$ is at least z. However, during the grouping phase, some of the clusters produced by the k-mode algorithm may be smaller than the threshold. These clusters are identified as *invalid sets*. Following the *Grouping Phase*, pseudonyms assigned to *invalid sets* will be kept until the protocol receives a new batch. The protocol groups these delayed pseudonyms with the new batch's pseudonyms. When a *valid set* (i.e., a set with a size larger than or equal to z) contains pseudonyms from different batches, the users of the pseudonyms in this set should publish under new pseudonyms during the *Communication Phase*. That is crucial to prevent $\mathcal{A}$ from partitioning users belonging to the same set into subsets or even de-anonymizing some of them using a timing attack based on when the pseudonyms began to receive messages.

Leaving the system. Most microblogging systems allow users to delete their accounts/pseudonyms when they no longer want to use the system. However, stopping the use of the system causes churn in the indistinguishability sets. Therefore, to ensure that users are indistinguishable after leaving the system, users must first inform the system that they wish to delete their pseudonyms. The users need to stick to the schedule until the protocol notifies them that their pseudonyms have been deleted. Once they receive this notification, they can stop using the system . The protocol deletes the user's pseudonym when there are at least a certain number of other users in the set who also want to delete their pseudonyms.

6 RELATED WORK

Anonymous microblogging systems. Several anonymity systems have been proposed throughout the last years to support the microblogging scenario. In these systems, sender anonymity is achieved using various techniques such as *mixnets* (Atom [20]), *DCnets* (Dissent [5]), *private information retrieval* (Blinder[1], Riposte [4], 2PPS [12], Spectrum [26]), and *random forwarding* (AnonPubSub [9]). For receiver anonymity, most of the proposed systems depend on the concept of broadcasting messages to all users [1, 4, 5, 20, 26], which results in high network overhead. Since broadcasting is not suitable for users with limited bandwidth, systems

like [9, 12, 15, 21] have addressed this issue by enabling anonymous multicast communication.

Traffic-analysis attacks. There are several types of traffic analysis attacks, such as timing attacks, intersection attacks, and statistical disclosure attacks. Tor, which is by far the most widely used anonymity system, is vulnerable to the three previously mentioned attacks [6, 8, 19]. Signal, a popular privacy-preserving instant messaging application, is also susceptible to a statistical disclosure attack that can effectively deduce the relationship between the sender and the recipient of an end-to-end encrypted message stream in the application [23]. Although mixnets are typically known for their ability to withstand traffic analysis attacks, there are studies [7, 19, 32] that evinced the effectiveness of statistical disclosure attacks against mix-based systems, especially when they support full bidirectional communications [7]. On anonymous email networks, statistical disclosure attacks based on the Expectation-Maximization algorithm were successful as well [29]. In [11], intersection attacks were demonstrated to be extremely effective on anonymous microblogging.

Mitigation techniques. Sending cover traffic throughout the whole mix network has been shown to be able to prevent some traffic analysis attacks. Nonetheless, it cannot overcome powerful attacks like statistical disclosure attacks, and intersection attacks [2, 11]. Many papers like [3, 5, 12, 14, 21] proposed protecting against intersection attacks by requiring all users who participate in the systems to have similar communication behavior, i.e., all users join the system at the same time, send at the same time slots, and have the same sending rates. Nevertheless, this requirement is not realistic. A method was proposed in [17] for forming possibilistic anonymity sets by grouping users based on their communication behavior. However, possibilistic anonymity sets do not ensure strong anonymity (i.e., indistinguishability) as it only ensures plausible deniability. Another solution was proposed in [33] to create anonymity sets that ensure indistinguishability. Nevertheless, this solution groups the users randomly into sets; hence, it is not efficient.

7 CONCLUSION & FUTURE WORK

In this paper, we propose a protocol for mitigating intersection attacks in anonymous microblogging systems. Our protocol addresses the unrealistic requirement in the literature that all users must commit to send messages all the time to prevent user de-anonymization via intersection attacks. It groups users based on their publishing behavior into sets. Then, for each set, it generates a communication schedule and requires users in the set to adhere to the schedule in order for them to appear indistinguishable from the point of view of an adversary. In our evaluation, we used real-world datasets from Twitter and Reddit to derive realistic user publishing behavior. We examined the users' behavior in these datasets and discovered that the majority of users in both datasets have low publishing rates. Our findings also show that scheduling can significantly reduce bandwidth overhead on the user's side. However, as expected, we have found that the reduction in the bandwidth overhead usually comes at the expense of latency. Therefore, the schedule for each set should be designed in such a way that it optimizes the trade-off between bandwidth overhead and latency based on the needs of the users in that set.

Future work should focus on testing our protocol on more datasets collected over a longer period. In addition, different ways of creating schedules can be applied and assessed. In particular, more sophisticated methods (e.g., machine learning algorithms) can be employed to design schedules based on predictions of user behavior. As a result, schedules might even be enhanced in terms of bandwidth and latency overhead. Furthermore, our mitigation protocol can be applied in other scenarios, such as anonymous messaging.

8 ACKNOWLEDGEMENTS.

This work was partially supported by funding from the German Research Foundation (DFG), research grant 317688284. We would like to thank Tim Grube for his insightful feedback on an earlier version of this work.

REFERENCES

[1] Ittai Abraham, Benny Pinkas, and Avishay Yanai. 2020. Blinder - Scalable, Robust Anonymous Committed Broadcast. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*. ACM, 1233–1252. https://doi.org/10.1145/3372297.3417261

[2] Oliver Berthold and Heinrich Langos. 2002. Dummy Traffic against Long Term Intersection Attacks. In *Privacy Enhancing Technologies, Second International Workshop, PET 2002, San Francisco, CA, USA, April 14-15, 2002, Revised Papers (Lecture Notes in Computer Science, Vol. 2482)*. Springer, 110–128. https://doi.org/10.1007/3-540-36467-6_9

[3] David Chaum, Debajyoti Das, Farid Javani, Aniket Kate, Anna Krasnova, Joeri de Ruiter, and Alan T. Sherman. 2017. cMix: Mixing with Minimal Real-Time Asymmetric Cryptographic Operations. In *Applied Cryptography and Network Security - 15th International Conference, ACNS 2017, Kanazawa, Japan, July 10-12, 2017, Proceedings (Lecture Notes in Computer Science, Vol. 10355)*. Springer, 557–578. https://doi.org/10.1007/978-3-319-61204-1_28

[4] Henry Corrigan-Gibbs, Dan Boneh, and David Mazières. 2015. Riposte: An Anonymous Messaging System Handling Millions of Users. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*. IEEE Computer Society, 321–338. https://doi.org/10.1109/SP.2015.27

[5] Henry Corrigan-Gibbs and Bryan Ford. 2010. Dissent: accountable anonymous group messaging. In *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS 2010, Chicago, Illinois, USA, October 4-8, 2010*. ACM, 340–350. https://doi.org/10.1145/1866307.1866346

[6] George Danezis. 2003. Statistical disclosure attacks: Traffic confirmation in open environments. In *Security and Privacy in the Age of Uncertainty: IFIP TC11 18 th International Conference on Information Security (SEC2003) May 26–28, 2003, Athens, Greece 18*. Springer, 421–426.

[7] George Danezis, Claudia Díaz, and Carmela Troncoso. 2007. Two-Sided Statistical Disclosure Attack. In *Privacy Enhancing Technologies, 7th International Symposium, PET 2007 Ottawa, Canada, June 20-22, 2007, Revised Selected Papers (Lecture Notes in Computer Science, Vol. 4776)*. Springer, 30–44. https://doi.org/10.1007/978-3-540-75551-7_3

[8] George Danezis and Andrei Serjantov. 2004. Statistical Disclosure or Intersection Attacks on Anonymity Systems. In *Information Hiding, 6th International Workshop, IH 2004, Toronto, Canada, May 23-25, 2004, Revised Selected Papers (Lecture Notes in Computer Science, Vol. 3200)*. Springer, 293–308. https://doi.org/10.1007/978-3-540-30114-1_21

[9] Jörg Daubert, Mathias Fischer, Tim Grube, Stefan Schiffner, Panayotis Kikiras, and Max Mühlhäuser. 2016. AnonPubSub: Anonymous publish-subscribe overlays. *Comput. Commun.* 76 (2016), 42–53. https://doi.org/10.1016/j.comcom.2015.11.004

[10] Euronews. 2022. Turkey-Syria earthquakes: How Twitter has helped find survivors trapped beneath the rubble. https://www.euronews.com/next/2023/02/10/how-twitter-helped-find-survivors-trapped-beneath-rubble-after-turkeys-earthquakes.

[11] Sarah Abdelwahab Gaballah, Lamya Abdullah, Minh Tung Tran, Ephraim Zimmer, and Max Mühlhäuser. 2022. On the Effectiveness of Intersection Attacks in Anonymous Microblogging. In *Secure IT Systems - 27th Nordic Conference, NordSec 2022, Reykjavic, Iceland, November 30-December 2, 2022, Proceedings (Lecture Notes in Computer Science, Vol. 13700)*. Springer, 3–19. https://doi.org/10.1007/978-3-031-22295-5_1

[12] Sarah Abdelwahab Gaballah, Christoph Coijanovic, Thorsten Strufe, and Max Mühlhäuser. 2021. 2PPS - Publish/Subscribe with Provable Privacy. In *40th International Symposium on Reliable Distributed Systems, SRDS 2021, Chicago, IL,*

USA, September 20-23, 2021. IEEE, 198–209. https://doi.org/10.1109/SRDS53918.2021.00028

[13] GeeksforGeeks. 2023. Elbow Method for optimal value of k in KMeans. https://www.geeksforgeeks.org/elbow-method-for-optimal-value-of-k-in-kmeans/.

[14] Nethanel Gelernter, Amir Herzberg, and Hemi Leibowitz. 2017. Two Cents for Strong Anonymity: The Anonymous Post-office Protocol. In *Cryptology and Network Security - 16th International Conference, CANS 2017, Hong Kong, China, November 30 - December 2, 2017, Revised Selected Papers (Lecture Notes in Computer Science, Vol. 11261)*. Springer, 390–412. https://doi.org/10.1007/978-3-030-02641-7_18

[15] George Giakkoupis, Rachid Guerraoui, Arnaud Jégou, Anne-Marie Kermarrec, and Nupur Mittal. 2015. Privacy-Conscious Information Diffusion in Social Networks. In *Distributed Computing - 29th International Symposium, DISC 2015, Tokyo, Japan, October 7-9, 2015, Proceedings (Lecture Notes in Computer Science, Vol. 9363)*, Yoram Moses (Ed.). Springer, 480–496. https://doi.org/10.1007/978-3-662-48653-5_32

[16] Tim Grube, Markus Thummerer, Jörg Daubert, and Max Mühlhäuser. 2017. Cover Traffic: A Trade of Anonymity and Efficiency. In *Security and Trust Management - 13th International Workshop, STM 2017, Oslo, Norway, September 14-15, 2017, Proceedings (Lecture Notes in Computer Science, Vol. 10547)*, Giovanni Livraga and Chris J. Mitchell (Eds.). Springer, 213–223. https://doi.org/10.1007/978-3-319-68063-7_15

[17] Jamie Hayes, Carmela Troncoso, and George Danezis. 2016. TASP: Towards Anonymity Sets that Persist. In *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society, WPES@CCS 2016, Vienna, Austria, October 24 - 28, 2016*. ACM, 177–180. http://dl.acm.org/citation.cfm?id=2994635

[18] Zhexue Huang. 1998. Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Min. Knowl. Discov.* 2, 3 (1998), 283–304. https://doi.org/10.1023/A:1009769707641

[19] Dogan Kesdogan, Dakshi Agrawal, and Stefan Penz. 2002. Limits of Anonymity in Open Environments. In *Information Hiding, 5th International Workshop, IH 2002, Noordwijkerhout, The Netherlands, October 7-9, 2002, Revised Papers (Lecture Notes in Computer Science, Vol. 2578)*. Springer, 53–69. https://doi.org/10.1007/3-540-36415-3_4

[20] Albert Kwon, Henry Corrigan-Gibbs, Srinivas Devadas, and Bryan Ford. 2017. Atom: Horizontally Scaling Strong Anonymity. In *Proceedings of the 26th Symposium on Operating Systems Principles, Shanghai, China, October 28-31, 2017*. ACM, 406–422. https://doi.org/10.1145/3132747.3132755

[21] Albert Kwon, David Lazar, Srinivas Devadas, and Bryan Ford. 2016. Riffle: An Efficient Communication System With Strong Anonymity. *Proc. Priv. Enhancing Technol.* 2016, 2 (2016), 115–134. https://doi.org/10.1515/popets-2016-0008

[22] Dong Lin, Micah Sherr, and Boon Thau Loo. 2016. Scalable and Anonymous Group Communication with MTor. *Proc. Priv. Enhancing Technol.* 2016, 2 (2016), 22–39. https://doi.org/10.1515/popets-2016-0003

[23] Ian Martiny, Gabriel Kaptchuk, Adam J. Aviv, Daniel S. Roche, and Eric Wustrow. 2021. Improving Signal's Sealed Sender. In *28th Annual Network and Distributed System Security Symposium, NDSS 2021, virtually, February 21-25, 2021*. The Internet Society. https://www.ndss-symposium.org/ndss-paper/improving-signals-sealed-sender/

[24] Karissa McKelvey and Filippo Menczer. 2013. Design and prototyping of a social media observatory. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013, Companion Volume*. International World Wide Web Conferences Steering Committee / ACM, 1351–1358. https://doi.org/10.1145/2487788.2488174

[25] Karissa Rae McKelvey and Filippo Menczer. 2013. Truthy: enabling the study of online social networks. In *Computer Supported Cooperative Work, CSCW 2013, San Antonio, TX, USA, February 23-27, 2013, Companion Volume*. ACM, 23–26. https://doi.org/10.1145/2441955.2441962

[26] Zachary Newman, Sacha Servan-Schreiber, and Srinivas Devadas. 2022. Spectrum: High-bandwidth Anonymous Broadcast. In *19th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2022, Renton, WA, USA, April 4-6, 2022*. USENIX Association, 229–248. https://www.usenix.org/conference/nsdi22/presentation/newman

[27] Andreas Pfitzmann and Marit Hansen. 2010. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management.

[28] Twitter Developer Platform. [n. d.]. Counting characters. https://developer.twitter.com/en/docs/counting-characters.

[29] Javier Portela, Luis Javier García-Villalba, Alejandra Guadalupe Silva Trujillo, Ana Lucila Sandoval Orozco, and Tai-Hoon Kim. 2016. Disclosing user relationships in email networks. *J. Supercomput.* 72, 10 (2016), 3787–3800. https://doi.org/10.1007/s11227-015-1524-7

[30] Statista Statistics. 2023. Most popular social networks worldwide as of January 2023, ranked by number of monthly active users. https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/.

[31] The Center for Strategic and International Studies (CSIS). 2022. Protest, Social Media, and Censorship in Iran. https://www.csis.org/analysis/protest-social-media-and-censorship-iran.

[32] Carmela Troncoso, Benedikt Gierlichs, Bart Preneel, and Ingrid Verbauwhede. 2008. Perfect Matching Disclosure Attacks. In *Privacy Enhancing Technologies, 8th International Symposium, PETS 2008, Leuven, Belgium, July 23-25, 2008, Proceedings (Lecture Notes in Computer Science, Vol. 5134)*. Springer, 2–23. https://doi.org/10.1007/978-3-540-70630-4_2

[33] David Isaac Wolinsky, Ewa Syta, and Bryan Ford. 2013. Hang with your buddies to resist intersection attacks. In *2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13, Berlin, Germany, November 4-8, 2013*. ACM, 1153–1166. https://doi.org/10.1145/2508859.2516740

P5

"IT'S NOT MY DATA ANYMORE": EXPLORING NON-USERS' PRIVACY PERCEPTIONS OF MEDICAL DATA DONATION APPS

This chapter was first published as

Sarah Abdelwahab Gaballah, Lamya Abdullah, Ephraim Zimmer, Sascha Fahl, Max Mühlhäuser, and Karola Marky. " "It's Not My Data Anymore": Exploring Non-Users' Privacy Perceptions of Medical Data Donation Apps." Proceedings on Privacy Enhancing Technologies 1 (2025): 654–670.

Contribution Statement: I led the idea generation, conceptualization, and development of the proposed work, including the study design, data analysis, and writing of the publication. Karola Marky assisted with the development of the study methodology and interview questions. Ephraim Zimmer contributed to the analysis of the sketches, while Lamya Abdullah helped with the analysis of the interview transcripts. Sascha Fahl and Max Mühlhäuser provided valuable discussions and insights on key aspects of the study. All co-authors contributed to the creation of the publication.

"It's Not My Data Anymore": Exploring Non-Users' Privacy Perceptions of Medical Data Donation Apps

Sarah Abdelwahab Gaballah
Ruhr University Bochum
sarah.gaballah@rub.de

Lamya Abdullah
Technical University of Darmstadt
abdullah@tk.tu-darmstadt.de

Ephraim Zimmer
Technical University of Darmstadt
zimmer@privacy-trust.tu-darmstadt.de

Sascha Fahl
CISPA Helmholtz Center for Information Security
sascha.fahl@cispa.de

Max Mühlhäuser
Technical University of Darmstadt
max@tk.tu-darmstadt.de

Karola Marky
Ruhr University Bochum
karola.marky@rub.de

Abstract

This paper contributes an in-depth investigation (N=24) of privacy perceptions in the context of medical data donation apps. *Medical data donation* refers to the act of voluntarily sharing medical data with research institutions, which plays a crucial role in advancing healthcare research and personalized medicine. To design effective medical data donation apps, we need to understand how privacy expectations affect people's willingness to use such apps. We focus on non-users—those who have no experience with medical data donation apps—because gaining a deeper understanding of their perceptions is essential for fostering the adoption of these apps. Our findings highlight the importance of trust, transparency, and anonymity as driving factors. Participants expressed a willingness to share highly sensitive medical data with the apps if they were assured of complete anonymity, yet criticism regarding the risks of de-anonymization was also raised. Based on our results, we identify privacy awareness issues, especially concerning data sensitivity. Additionally, we explain the differences between participants' privacy expectations and preferences and what existing medical data donation apps offer. Finally, we provide guidance for the development of future user-centric medical data donation apps.

Keywords

privacy, anonymity, mental models, medical data donation apps, trust, transparency, control, data sensitivity

1 Introduction

Medical data donation is the voluntary act of providing health-related information for research initiatives and medical databases [9]. Health information includes any personal data related to an individual's past, current, or future physical or mental health [20], and may also cover fitness data (e.g., heart rate, respiratory rate, blood oxygen levels) that can reveal health issues [63]. Donating such data enhances early disease detection and understanding of diseases, ultimately leading to better treatments [22]. Several research institutes have created apps to gather health data from people for various research purposes. One notable example is the Corona-Data-Donation-App (CDA) in Germany, which was launched during the COVID-19 pandemic. Although more than 500,000 users downloaded the CDA app, indicating a willingness among individuals to share their data for medical research [52], studies revealed significant resistance among many to share their data, primarily due to security and privacy concerns [14, 29, 60, 67].

Protecting the privacy of medical data donors is not only an ethical imperative but also ensures their trust and willingness to share their data [18, 64]. Given that donated data is typically highly personal and can reveal significant sensitive information about the donors' health, linking donors to their data could lead to risks such as discrimination. Therefore, it is essential in this context to extend privacy protections to guarantee anonymity [61]. Since the goal of data donation is to identify broad patterns rather than diagnose individuals, concealing the identities of data donors should not affect the research analysis and results.

Several studies investigated privacy concerns and perceptions related to sharing medical data (cf. [4, 13, 32, 37, 51, 65, 74]). However, inconsistencies exist among these studies regarding the level of trust in researchers, the understanding of potential privacy risks and protection methods, and the effectiveness of the privacy measures in encouraging data sharing. Although research highlights the importance of anonymity for facilitating medical data sharing [13, 64, 65], there has been limited exploration into how people understand anonymity and misconceptions they might have. Furthermore, there is a lack of understanding about how people's misconceptions or lack of awareness regarding privacy and anonymity might influence their willingness to use medical data donation apps.

To address these gaps in existing research, we conducted a study focusing on perceptions and understanding of privacy in the context of medical data donation apps, with an in-depth investigation of anonymity. Given the limited use of these apps, we explored the perceptions and expectations of non-users—those who never used a medical data donation app. By gaining insight into their perceptions and identifying potential obstacles or misunderstandings, we can design apps that address these issues, making them more appealing and accessible, thereby promoting greater usage. Our research specifically considers the following main research question:

RQ: What are privacy perceptions and expectations of non-users in the context of medical data donation apps?

Proceedings on Privacy Enhancing Technologies 2025(1), 654–670

https://doi.org/10.56553/popets-2025-0035

For this, we conducted semi-structured interviews (N=24) where we asked participants to participate in a drawing exercise. Based on the drawings, we discussed the participants' wishes, expectations, perceptions, and *speculative* mental models, i.e., how non-users imagine the usage of a data donation app. From our results, we learned that most participants struggled to illustrate a detailed mental model of their expected data donation infrastructure. We also found that participants trust data donation apps that are not driven by commercial gains and are provided by research institutes. The participants expressed a desire to control what they share and how it is used but feared that this control might burden them with technical and medical complexities. Further, they were concerned about data breaches, misuse, and discrimination, but had limited understanding of how these risks could occur. They wanted strong privacy guarantees from medical data donation apps and insisted on anonymity, expressing unwillingness to use these apps if their data could be linked to their identities or locations. However, they had awareness issues about the sensitivity of the data that these apps could collect and the methods that can ensure the privacy and anonymity of data donors. When we compared participants' speculative mental models with two existing medical data donation apps, we found significant gaps that might hinder user adoption. To develop user-friendly medical data donation apps that align with privacy and anonymity expectations, we propose several key design recommendations.

Research contributions: In the course of this paper, we make the following contributions:

(1) **First mental model investigation of medical data donation apps:** We present the first investigation of perceptions of medical data donation apps. We specifically investigated the speculative privacy mental models of 24 participants (non-users) through semi-structured interviews and a drawing exercise.
(2) **Analysis of perceived expectations, risks & misconceptions:** Among our results, we show expectations regarding data collection, storage, and access in medical data donation apps and highlight perceived risks, protection measures, and misconceptions regarding privacy and anonymity.
(3) **Comparison of users' mental models to existing apps:** We compare participants' expectations regarding privacy and anonymity guarantees to the protection measures implemented by two well-known existing data donation apps proposed during the COVID-19 pandemic.
(4) **Overall recommendations for human-centered medical data donation apps:** We conclude with recommendations for research institutes regarding how to design usable medical data donation apps that meet users' needs and expectations.

2 Background & Related Work

This section provides an overview of privacy, anonymity, medical data donation apps, mental models, and a summary of related work.

2.1 Privacy & Anonymity

Privacy grants individuals the ability to prevent involuntary disclosure by affording them the right to protect personal information across various contexts. It is commonly understood as both an individual mechanism for revealing and concealing aspects of oneself and a contextual norm governing information flows, e.g., who has access to what information [43]. Early research on Privacy Enhancing Technologies (PETs) categorized privacy into four main areas: 'freedom from intrusion', 'negotiating the public/private divide', 'identity management', and 'surveillance' [50].

Anonymity, as a means for enhancing privacy, is linked to control over identity management and surveillance. It is typically regarded not only as a way to protect identity information but also to withhold it entirely [44]. Achieving anonymity involves concealing multiple dimensions of identity knowledge, including legal name, location, behavior patterns, and personal characteristics [42].

2.2 Privacy & Anonymity Techniques for Medical Data Donation

Several techniques have been proposed in the literature to protect users' privacy and anonymity when sharing their medical data. One common technique is pseudonymization, which involves removing personal identifiers from data and replacing them with placeholder values (i.e., pseudonyms). However, this technique proves ineffective when it comes to protecting data against a wide range of re-identification threats [34]. For example, if an individual's record is unique based on information like age, job, sex, or ZIP code, an attacker with this information can directly link the record to its owner [61]. There are other techniques that provide better protection by altering personally identifiable information (PII), both direct and indirect, within data to prevent the linkage of individuals to specific data points. Common examples of such techniques include generalization, suppression, k-anonymity, and differential privacy [15, 24]. Generalization reduces data granularity by replacing specific values with more generalized ones, while suppression selectively removes sensitive information to preserve privacy. k-anonymity ensures that each record in a dataset is indistinguishable from at least k-1 others. Differential privacy (DP) protects privacy by adding noise to data; this noise can be introduced by users (local DP) or by a data aggregator. Additional techniques for privacy-preserving medical data donation extend protection beyond data to also prevent linking users to their shared data based on communication metadata, e.g., IP addresses. Such techniques are often based on secure multi-party computation or secret-sharing-based methods [26].

2.3 Medical Data Donation Apps

Over the past few years, many apps for donating medical data have been introduced. These apps can collect similar data to fitness apps, yet differ from them in their data collection purposes, user consent, and methods. Data donation apps gather data for research purposes, with users willingly and fully informed about that. In contrast, fitness apps collect data for user health monitoring, but users may be unaware that their data is shared with other third parties, e.g., for advertising and marketing purposes [28]. Additionally, data in fitness apps is typically collected only through trackers, whereas medical data donation apps might combine trackers and questionnaires.

Medical data donation apps vary in the types of data they collect, but they can have similar infrastructure and privacy protections across contexts (COVID or non-COVID). However, motivation to participate tends to be stronger during critical situations like the COVID pandemic, as individuals are often more driven to contribute to efforts aimed at managing the crisis [14]. It is important to note that medical data donation apps differ from COVID contact-tracing apps in their goals and methods. Data donation apps focus on collecting health information to create datasets for research to improve disease understanding and treatment. Conversely, contact-tracing apps are designed for real-time contact tracing, using technologies like Bluetooth or GPS to alert users about potential COVID-19 exposure and help prevent the virus's spread [1].

In our study, we presented the Corona-Data-Donation (CDA) and SafeVac apps as examples of data donation apps because they are the most recognized German apps in this field, given that the study was conducted in Germany.

Corona-Data-Donation-App (CDA). The Corona-Data-Donation-App was developed by the Robert Koch Institute (RKI) with the aim of collecting data from users for purposes, such as detecting COVID-19 symptoms and constructing fever maps [52]. Users are required to provide health-related information, including symptoms experienced, vaccination status, and any pre-existing medical conditions. Also, users are requested to link their wearable devices to the app to allow for the collection of data such as temperature, heart rate, sleep patterns, and activity levels. Additionally, the app requires demographic information (e.g., age, gender, and location) and contact details (e.g., email address or phone number). Pseudonymization is employed to safeguard user privacy. However, this may not be sufficient to protect sensitive data in the event of a breach, particularly considering the identifiable information collected, such as location, email address, and phone number. Further, since users send their data directly to the RKI, the institute can potentially link users to their donated data through IP addresses. Moreover, the app is susceptible to several other privacy and security risks, as discussed by Tschirsich et al. [41].

SafeVac. The Paul-Ehrlich Institute (PEI) developed this app to study the effects of COVID-19 vaccines [17]. The app collects demographic data (age, weight, height, and gender), vaccination details, and health status (e.g., pre-existing medical conditions and current medications). Additionally, it prompts users to complete questionnaires at specific intervals after vaccination. These questionnaires are used to track adverse events by recording symptoms and their impact, as well as to gather feedback on the vaccination experience and follow-up actions. Pseudonymization is also implemented in this app to protect user privacy [2]. Unlike CDA, where data is sent directly to the RKI, SafeVac uses a government server as an intermediary between the PEI and users [2]. This server receives data from users and forwards it to the PEI, thereby preventing the PEI from identifying which user sent the data. While SafeVac offers greater privacy compared to CDA, it still relies on pseudonymization, rendering it susceptible to re-identification attacks [48].

2.4 User Attitudes and Awareness of Sharing Health-Related Information

According to many studies [5, 54, 66, 74], users' willingness to share their fitness data is influenced by their understanding of how the data is used and the benefits, if any, they receive from sharing it. Another common finding is that users are more likely to share their fitness data when they believe the benefits outweigh the potential risks and the recipient (e.g., service providers, third-party apps, and individuals) will use the shared data positively [4, 51, 60, 67], referring to the privacy calculus model [19]. Regarding donating medical data for research, several studies [4, 13, 14, 27, 51, 58–60, 67] found that participants generally have a positive attitude, and their beliefs in the benefits of medical research strongly motivate the willingness to donate data. Research shows that people mainly donate their data for altruistic reasons, such as supporting research and advancing healthcare for the benefit of society and future patients [7, 13, 18, 25, 45, 55, 59], though some are also driven by monetary incentives [37, 55, 59, 65]. Brown et al. [13] found that participants did not identify health-related stigma as a barrier to sharing their personal health data. Additionally, in a study by Seltzer [58], the majority of participants expressed interest in receiving the results of analyses conducted on their shared data, with half of them expressing a desire for their healthcare provider to be informed about the results as well.

Valdez and Ziefle's study [65] found that people were hesitant to share data about mental health but were more comfortable sharing information about physical health. Brown et al. [13] also found that participants were very cautious about sharing information related to sexuality. Additionally, in a study by Belen-Saglam et al. [8], participants exhibited a significant resistance to disclose information they considered irrelevant or out of context. Garrison et al. [27] discovered that individuals who were concerned about privacy and confidentiality were less likely to share their data. Further, both Garrison et al. [27] and Buhr et al. [14] observed less willingness to share data among minority groups. People generally preferred sharing their data with academic researchers rather than with businesses [27, 51, 65], government databases, or pharmaceutical companies [27].

Regarding privacy awareness, Zufferey et al. [74] found that many users of wearable activity trackers were aware of the privacy implications of sharing their fitness data, contrasting with Alqhatani et al.'s [5] findings where most users were unaware of privacy risks. Richter et al. [51] reported that about 70% of their study participants trusted medical researchers to handle their data responsibly. However, in other studies, including by Voigt et al. [67], Sleigh et al. [60], and Aitken et al. [4], participants expressed concerns about researchers potentially misusing the data. Aitken et al. [4] identified worries regarding confidentiality and users' control over their data, along with low awareness of users about current data privacy practices.

There are few studies that have investigated the impact of privacy and anonymity on individuals' decisions regarding medical data sharing. Kacsmar et al. [32] examined the impact of five privacy and anonymity techniques on user acceptability. However, the results indicated that participants had a very low level of understanding regarding these techniques. Also, Kühtreiber et al.[35] found similar

results, as participants were not able to fully comprehend differential privacy. Valdez and Ziefle [65] studied two anonymization techniques, k-anonymity, and differential privacy, and found that anonymity was the most important factor for participants in their study, regardless of the used anonymization technique. Brown et al.[13] discovered that offering the option to remain anonymous encourages individuals to share health data within online communities. Contrarily, Belen-Saglam et al. [8] did not find anonymity to be of significant importance. Their study revealed that, with the exception of data related to sex lives, participants generally did not prioritize anonymity when sharing health data. The researchers suggested that this lack of emphasis on anonymity may be due to the fact that their participants were all from the UK, where there is considerable trust in the national health service.

2.5 Mental Models

Mental models are internal representations humans derive from the real world to use a technical system [30]. This can have various levels of details that differ between humans [11, 30, 33, 68, 70]. Overall, there are two main types of mental models: functional and structural models [46]. Users with functional models know how to use a system but do not understand how it works in detail. Users with structural models have a thorough understanding of how the system works. Consequently, having a mental model requires some interaction with a system. This paper investigates *speculative* mental models, which are the users' internal representations of a system they have not used yet.

Misconceptions in mental models may lead users to engage in behaviors that do not always reflect their true needs. Thus, the mental models must be sound enough for users to interact with technology effectively [36].

Privacy concerns and misconceptions have repeatedly been shown to impact the usage intention of IoT devices [3, 62, 71, 73], or digital health records [6, 49]. The majority of related work focused on privacy as a rather generic concept in the context of mental models. The solutions proposed in most papers centered on raising awareness [71, 73], enabling control [62], or education [3]. This, however, comes with several challenges considering digitization as a whole because we likely do not have the capacity to educate individuals in-depth about each and every aspect of each system to create structural mental models. Inspired by these related studies on mental models and their findings, we decided to further investigate the medical data donation domain as a special use case where data must not be linked to the identities of individuals, consequently demanding the highest level of privacy— which is anonymity.

Summary. Related work suggests that people generally have a positive attitude toward data sharing for medical research. However, there are inconsistencies among the findings of existing studies regarding awareness levels of privacy risks and protections. There is also limited knowledge about how non-users perceive data donation apps and the sensitive nature of the data they collect. Additionally, there has been insufficient exploration of how perceptions of anonymity influence the willingness to use and engage with medical data donation apps. To increase the adoption of these underused apps, we address these gaps and contribute to the literature by examining the privacy expectations, understanding, and speculative mental models of non-users, with a particular focus on anonymity. To achieve this, we gathered users' perceptions using both drawings and conversation-based interviews. In contrast, relevant studies have primarily relied on either surveys or conversation-based interviews to capture users' perceptions.

3 Methodology

We conducted an interview study with 24 participants to answer our research question. Our study consisted of two parts: 1) a drawing exercise where participants were asked to explain and sketch their mental models; and 2) a semi-structured interview to delve deeper into their understanding. We chose to include drawing exercises because they are effective in capturing users' mental models of specific systems or technologies [31]. We used semi-structured interviews due to their balance of structure and flexibility in exploring participants' perceptions in depth [47].

Participants and Recruitment. We did our study in Germany, where the adoption of medical data donation is very low. We recruited 24 participants who were non-users of data donation apps. We utilized various methods, such as mailing lists, flyers, poster advertisements, social networks, and word-of-mouth, to reach out to potential volunteers. All participants were required to be at least 18 years of age. Thirteen participants identified as men, ten as women, and one as non-binary. The average age of all participants was 29.37 years (SD=10.68, Min=19, Max=65). The distribution of the participants' ages reveals the following frequency counts within 10-year ranges: 1 individual [10-19], 14 [20-29], 6 [30-39], 1 [40-49], 1 [50-59], 1 [60-69].

Eleven of the participants attended school or university. Ten were employed full-time, and one participant was retired. Two participants identified themselves as housewives. There was a variation in the educational levels: nine individuals had a high school diploma, and five had a bachelor's degree. Ten had advanced degrees: one participant held a PhD, and nine had a master's degree. An overview of our sample is presented in Table 1. We used the ATI scale [23], which ranges from 1 to 6, to determine participants' affinity for technology. A higher score indicates a greater affinity for technology. Our sample had an average ATI score of 4.09 (minimum = 3, maximum = 5.11, SD = 0.66), which suggests that the participants have a high affinity for technology [23]. To assess participants' privacy perception, we considered the 10-item IUIPC questionnaire [38]. Overall, they rated their desire for control at a mean of 5.81, their awareness of privacy practices at a mean of 6.30, and the perceived ratio between collection and benefits at a mean of 5.96. That indicates that participants were more concerned about their privacy as they had high scores on the IUIPC scale. For more detailed values, see Table 1.

Study Procedure. The session we had with each participant consisted of five main parts. All the questions asked to participants are included in the Appendix A.3. The sessions were audio-recorded, with the drawing process being video-recorded as well. Each session lasted about an hour in total. The detailed procedure is as follows:

1) Consent & Demographics. Participants were first informed of their rights, and the collected data, and that they could end the study at any time without any negative consequences. Additionally,

ID	Age	Gender	Education	Job	Study Field	ATI Scale	IUIPC			Spec. Mental Model
							Control	Awareness	Collection	
P1	31	Woman	PhD	Researcher	Psychology	5.11	7	7	6.5	Intermediate understanding
P2	23	Woman	B.Sc.	M.Sc. Student	Industrial eng.	5	6.67	7	6	Advanced understanding
P3	23	Male	High school	B.Sc. Student	Informatics	4.89	5.67	5.33	3.75	Advanced understanding
P4	19	Woman	High school	B.Sc. Student	Informatics	4.67	5.67	5.34	5	Advanced understanding
P5	23	Man	High school	B.Sc. Student	Informatics	3.44	7	6.67	6	Advanced understanding
P6	28	Man	M.Sc.	Research Associate	Informatics	4.78	5.67	5.67	4	Advanced understanding
P7	22	Man	High school	B.Sc. Student	Informatics	4.11	5.67	6	6.75	Intermediate understanding
P8	20	Man	High school	B.A. Student	Cognitive science	4.11	4	6.33	7	Intermediate understanding
P9	30	Woman	High school	B.Sc. Student	Informatics	3	6	6.67	6.5	Intermediate understanding
P10	31	Woman	M.Sc.	Software engineer	Informatics	3	3	5.33	7	Misconception-based understanding
P11	29	Man	B.A.	M.A. Student	Psychology	4	7	5.33	7	Intermediate understanding
P12	25	Man	M.Sc.	Software engineer	IT-Security	4.78	6	7	7	Advanced understanding
P13	30	Man	High school	B.Sc. Student	Cognitive science	4.56	6.33	6.67	6	Advanced understanding
P14	42	Woman	M.Sc.	Engineer	Electronics eng.	3.89	5.33	7	7	Intermediate understanding
P15	32	Woman	M.Sc.	Engineer	Architectural eng.	4	6.33	6.67	5.25	Misconception-based understanding
P16	21	Man	High school	B.Sc. Student	Civil eng.	4.56	4.67	6.33	5.25	Intermediate understanding
P17	30	Man	M.A.	Admin. Specialist	Management	4.11	6.67	7	6	Misconception-based understanding
P18	26	Man	B.Sc.	Software engineer	Informatics	3.44	6	6	5	Misconception-based understanding
P19	21	Man	High school	B.A. Student	Cognitive science	4.78	5.33	7	6	Intermediate understanding
P20	26	Nonbinary	M.Sc.	Research Associate	Bio-medical	3.22	5.33	5.67	7	Intermediate understanding
P21	65	Man	M.Sc.	Retired	Physics	3.11	6.33	6.33	5.75	Intermediate understanding
P22	54	Woman	B.A.	Housewife	Arts	4.22	5.67	6.67	6	Intermediate understanding
P23	25	Woman	B.A.	Housewife	Law	3.78	5.33	5.33	5.25	Misconception-based understanding
P24	29	Woman	M.A.	Research Associate	Economics	3.67	6.67	6.67	6	Intermediate understanding

Table 1: Participants' demographics, education, occupation, ATI scale, IUIPC scores, and mental models.

they were informed that the interview was audio-recorded and that the drawing exercise was filmed without their faces being captured. This, along with additional information about data and privacy protection ensured for participants, was provided to them in an information sheet, which also included a consent form. Participants were asked to read and sign the consent form. Following this, each participant provided demographic information, such as age, gender, education, and occupation. They also completed the questionnaires of the ATI scale [23] and the IUIPC scale [38].

2) Warm-Up & Anchoring. We asked the participants about their understanding of medical data donation and whether they had already heard about it. To make sure all participants have a common understanding, we explained our definition of medical data donation. Following that, we asked warm-up questions, such as whether they had ever donated their medical data, if they had any experience with medical data donation apps, and what situations or settings would encourage them to donate their data. We then introduced the following scenario: there is an app that lets users donate medical data to a research institute. The collected data is used by researchers in this research institute to better understand diseases and improve public health. To make the scenario more tangible for our participants, we provided them with two examples of data donation apps—specifically, the two apps explained in Section 2.3. In terms of the information shared with participants about these apps, we provided only the app's name, the research institute that developed it, its purpose, and the method used to collect data from users (SafeVac uses a questionnaire to gather data, while CDA retrieves data from users' fitness trackers). Then, we asked them if they had experience with any of the mentioned apps. For details regarding the participants' familiarity and prior experience with data donation and its apps, please see Table 3 in Appendix A.2.

3) Drawing Exercise. After that, the participants were requested to conduct the drawing exercise. We asked them to sketch their expectations of how a medical data donation app works. This involved illustrating the data flow and connections between various components. It's important to note that, while SafeVac and CDA were provided as examples, participants were not restricted to depicting the infrastructure of either one.

We provided the participants with paper in DIN A3 size and pens in different colors as recommended by related work [39, 72]. Research indicates that free-hand drawing without support can require excessive cognitive effort, which can be reduced by using cutout figures to make the drawing task easier [53, 56]. Therefore, we provided a wide range of printed cut-outs of several components, such as a user, a researcher, a research institute, a smartphone, a smartwatch, smart glasses, a questionnaire, the Internet, a server, a printer, a scanner, message, email, and router. Moreover, we explained that they do not need to use all the provided icons and they should only pick the ones that they wish to use in their sketch. During the drawing exercise, we encouraged participants to think aloud [10] and comment on what they were drawing, so that we could understand their thinking process. Several previous studies demonstrated the effectiveness of this combination [39, 71, 72]. We also asked the participants follow-up questions after finishing their drawing to ensure all drawn parts were explained in detail.

4) Semi-Structured Interview. We used the sketch from the previous part as the basis for the interview. We asked questions about storage, access, control, trust, privacy, and anonymity. To refine the interview script, we first conducted pilot interviews to identify and address issues with clarity of questions, structure, flow, timing, and participant comfort. Researchers took notes during these interviews, analyzed the feedback, and made necessary adjustments to the questions and procedures.

5) End & Reimbursement. Following the interview, we gave participants the opportunity to ask questions and provide additional feedback. Finally, each participant received ten euros as compensation, which was not subject to tax. This amount was consistent with Germany's minimum wage requirements: given that the minimum wage at the time of our study was €12.00 per hour pre-tax [57], ten

euros would be equivalent to or more than the after-tax earnings for one hour of work.

Ethical Considerations. In conducting our study, we adhered to the guidelines set by the ethics committees in the institutions of the authors. At our institutions, user studies must restrict the gathering of personal data to safeguard the participants' privacy. Every participant was given a random identifier. Before the interview, each participant signed a consent form, which was stored separately from all other information to ensure that it could not be linked to their identities. Prior to the study, we received approval from the ERB of the Technical University of Darmstadt.

Data Analysis. We ensured that all collected data was anonymized prior to the analysis. The audio transcripts were converted to text and personal information was replaced with neutral markers. To prevent participants from being identified through their handwriting in the drawn models, machine-generated text by a picture editing tool was used to conceal the handwriting. The sketches and interview transcripts were then analyzed in two parts using thematic analysis [12].

First, we analyzed the mental models expressed in the sketches. We ordered the sketches from undetailed to very detailed. Then, we used an open-coding approach with two authors serving as coders. By reviewing all sketches the two coders generated and agreed on a final codebook. The codebook consisted of four codes for the expressed level of detail. They then coded each sketch independently. The results were discussed, and the final code allocations for each drawing were decided. We considered the audio recordings throughout the analysis to supplement the information expressed in the sketches in cases where parts of the drawing were unclear.

Second, we analyzed the interview transcripts to capture the participants' mental models. We conducted open coding by assigning codes to meaningful and relevant concepts related to our research questions. One researcher, who conducted the interviews and was familiar with the data, proposed an initial codebook. A second researcher who was present during some of the interviews and had also reviewed the transcripts agreed on the final codebook in discussion with the first researcher. The final codebook consists of eight final categories of codes and 72 codes (see also Appendix A.1). One researcher followed the methodology guidelines for conducting thematic analysis and coded all statements using the codebook. The second researcher verified this, and any disagreements were resolved. It should be noted that thematic analysis guidelines advise against using double or multiple independent codings and relying on inter-rater reliability to demonstrate reliability [16]. This is because qualitative research acknowledges the researcher's influence on the process.

Limitations. Those are the limitations of our study: First, due to the qualitative nature of our study, we cannot make any quantitative conclusions. Also, our study relies on self-reported data and assessments, which might be biased due to social desirability, availability bias, and wrong recalls or self-assessments. As a result, our data only reflects the highly subjective perspectives of our participants. Additionally, we captured the speculative mental models of non-users, those might alter when using a data donation app. Further, we analyzed participants' mental models in relation to the CDA and SafeVac apps, given their relevance to the study location (Germany), which might also restrict the generalizability of the findings. Moreover, our results may be influenced by cultural bias. Therefore, the findings might reflect a perspective shaped by German cultural attitudes toward privacy, which are unique due to the country's history and privacy laws [69]. Research [14, 64] also shows that people in Germany generally have a positive attitude toward data donation for research purposes, which may differ from attitudes in other countries.

Finally, despite our efforts to recruit a diverse sample, our study may lack representativeness, as all participants had college-level education or higher, which typically indicates good knowledge, awareness, and cognitive skills. As a result, our findings might not generalize well to less-educated individuals. Nevertheless, our exploratory study provided an initial step in examining the speculative mental models of non-users of medical data donation apps. Future work should investigate a more representative sample.

4 Results

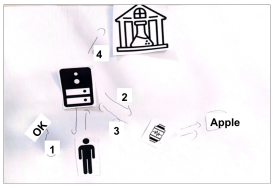
This section outlines the outcomes of our study. We begin by explaining the sketches. Then, we delve into the thematic analysis results, organized by themes. We offer quantifiers of mentions to give the reader an impression of how often a certain aspect was brought up, yet this is not an attempt to quantify our findings.

It's important to note that there is no singular, definitive infrastructure for medical data donation apps, which makes it challenging to establish a single ground truth. In our study, we chose to use CDA and SafeVac—the most notable medical data donation apps in Germany, where the study was conducted—as baselines and compared participants' expectations with these apps.

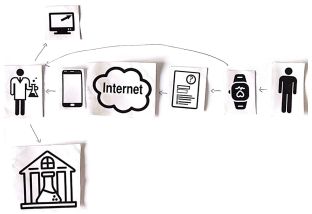
4.1 Level of Detail

Based on the sketches and the interviews, we found three types of speculative mental models about medical data donation apps, each with a different level of detail. Our classification process involved the evaluation of various factors: data flow, connections among different entities, the presence of key entities (such as data sources, storage, and data recipients like researchers or research institutes), the complexity of depicted entities, and incorporation of any security-related measures. To see each participant's mental model type, refer to Table 1.

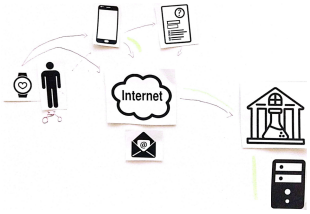
1) Misconception-based Understanding. Five participants illustrated an infrastructure for medical data donation that represents a misconception-based understanding. This model would form a functional mental model, given the provided details are limited [46]. The sketches created by these participants either portrayed an abstract data flow or failed to resemble that of a realistic data donation app. Some participants depicted only a few entities of the infrastructure (see Figure 1a). Connections between entities often deviated from real-world scenarios. Also, participants expressed difficulty in determining which entities should be connected to the internet. For instance, P23 connected the researcher directly to the user's smartphone and smartwatch, with no internet connection between the researcher and these devices, while an internet connection was established between the questionnaire and the smartphone (see Figure 1b).



(a) Sketch of P18 (Misconception-based Understanding)

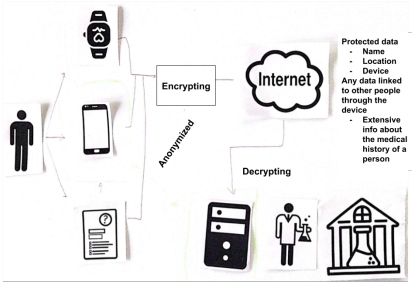


(b) Sketch of P23 (Misconception-based Understanding)

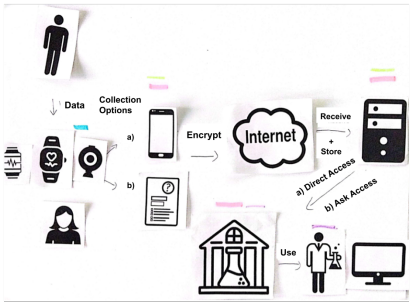


(c) Sketch of P8 (Intermediate Understanding)

(d) Sketch of P19 (Intermediate Understanding)



(e) Sketch of P2 (Advanced Understanding)



(f) Sketch of P6 (Advanced Understanding)

Figure 1: Examples of Participants' Sketches

All participants with this mental model were neither familiar with the term *data donation* nor had they experience with medical data donation or data donation apps. According to their scores on the ATI scale, three participants had medium technology affinity, while two had high technology affinity. As for their scores on the IUIPC scale, all participants scored 5 or higher in three categories: control, awareness, and collection, with the exception of one participant who scored 3 for control. These high scores indicate significant privacy concerns, a sentiment expressed also during the interviews. This shows that even individuals with basic knowledge can have strong worries or needs about privacy.

2) Intermediate Understanding. Twelve participants had this model type. They sketched the main components of a realistic infrastructure and connected these components in a realistic manner. Some of them also included multiple data sources, routers, and icons representing data transmission, processing, searching, and printing in their sketches. For examples, see the sketch of P8 (Figure 1c) and the sketch of P19 (Figure 1d). The level of understanding in this model, with its attention to detail and anticipation of data flow and connections, represents a structural model [46].

Most participants with this model showed a high affinity for technology interaction based on their scores on the ATI scale. Additionally, their IUIPC scores were high, indicating significant privacy concerns and awareness. Although participants with this model demonstrated a more sophisticated understanding compared to those with the basic understanding model, they were unfamiliar with the *data donation* term and lacked experience with related apps, except for two who mentioned some familiarity but hadn't used the CDA or SafeVac app.

3) Advanced Understanding. This model was demonstrated by seven participants. They provided very detailed sketches that demonstrated deep understanding. The illustrated components, their connections, and the depicted data flow reflect what could be realized in a realistic infrastructure. Unlike participants with other models, all participants with this model included security-related entities or measures in their sketches. For example, all of them used encryption to protect transmissions, and some participants also sketched other security safeguards, such as anonymity or access control. Refer to the sketch of P6 (Figure 1f), where the representation of data flow was comprehensive, integrating security measures like encryption and access control. Similarly, P2's sketch (Figure 1e) emphasized the necessity of encrypting data before transmission over the internet, followed by decryption at the server end. P2 also highlighted the importance of anonymizing communication between the smartphone and the server to hide the user's location. This model is a sophisticated structural model [46].

Most participants with this mental model had previously donated their medical data, but only in offline settings. Among them, only two had heard about CDA or SafeVac, and none had experience using these apps or medical data donation apps in general. During the interviews, these participants demonstrated a high understanding of privacy, anonymity, potential threats, and protection measures.

4.2 Motivations & Expectations

Participants were questioned about factors that could motivate their use of medical data donation apps and their expectations from such apps.

Diverse Motivations & Privacy-Related Barriers. We found that participants' motivations varied, which aligns with research on other domains like blood donation that demonstrates the multifaceted nature of prosocial motivation [21]. Ten participants expressed that their motivation to use medical data donation apps would be to contribute to the advancement of medical research, a sentiment consistent with findings from other studies on different data sharing scenarios [7, 18, 55, 59]. Another mentioned motivation was the desire to aid humanity and those in need, particularly during times of crisis (N=5), similar to findings in [7, 13, 18, 45, 59]. Sample comments from the participants include P2 stating, *"To help the research basically, because I think the more people participate, the better the results of the research will be"* and P1 commenting, *"If I see a benefit for society or for people in general not just for a company"*. Other motivations were getting a reward, personal benefits, or financial gain (N=3), similar to related work on the privacy calculus [19] and the studies in [55, 59, 65]. Also, three participants mentioned that if they were suffering from specific or rare diseases requiring further study, they would be motivated to use a data donation app that collects data on these conditions.

However, there are participants (N=2) who stated that nothing could persuade them to donate their data: they either had strict privacy needs or did not trust the researchers as they believed the researchers could misuse the data and not protect the participants' privacy as they promised. For example, P15 said: *"Nothing could encourage me because I'm not someone who would agree to share medical data. I'd rather keep my data private to safeguard my privacy. Even if I were assured of data protection, I would still decline to share my data"*.

Desired Data Control, Yet Perceived Infeasible. In line with other studies on data sharing [7, 14, 45, 64], participants (N=15) voiced expectations related to transparency. Their expectations included understanding how data would be used, collected, stored, and secured, as well as knowing who would have access to the data and when it would be deleted. Interestingly, we found that some participants (N=3) also expressed a desire to track the usage of their data. For example, P10 said, *"I need to know if the data will be used by only one organization or if it will be shared among several organizations or research institutions. I would like to be able to track where my data is stored and who has access to it"*. Similarly, P24 mentioned, *"I have concerns about how the researchers may use my data. They may use it for bad things, rather than for the benefit of society as they claim. If researchers give me complete information about their research and goals, as well as the ability to track what they do with my data and the results of their research, I will be willing to donate my data"*.

The transparency mentioned above enables participants to exert control. We found that participants (N=10) demanded control over the shared data, and also wanted to choose the studies that benefit from the data, e.g., P4: *"Because, it's my data, so I should have the right to choose which parts of it I want to share. Even if I can't immediately say which data I want to keep private, there might be still something I don't want to share. In that case, having the ability to decide not to donate that specific data would be important to me"*.

Even though the participants clearly wanted control, they had misconceptions about it. Some (N=3) believed that controlling what to donate and which studies could use it was infeasible. They saw only a binary choice: agree to all the terms and provide all requested data or decline participation entirely. Additionally, others (N=2) believed that if control options were provided, they would be unable to utilize them because it would require medical or technical expertise, which they, as normal users, lack. For example, P22 said: *"No, I do not need to have control because I am not a medical expert"*.

Research-Exclusive Use is Favored over Private Companies. Participants emphasized the importance of having confidence in the institute responsible for overseeing the data donation app. Factors contributing to trustworthiness include the institute's renown, positive reputation, official (governmental) status, or exclusive research focus. Similar to the findings of Buhr et al. [14], the majority of participants (N=19) in our study voiced limited trust in private institutes or companies, expressing negative opinions about sharing their medical data with them, e.g., P19 mentioned: *"I think we hear more scandals and problems about the private companies which have these data breaches or something, so surely I would not trust a data donation app from a private company"*. Participants were divided regarding data donation to apps operated by government institutes versus those managed by research institutes. While some (N=3) perceived government institutes to possess superior data protection capabilities, others (N=4) asserted that research institutes could offer better protection, with more commitment to data integrity and non-misuse.

Similar to findings in [13, 27, 51, 65], which show that participants are unwilling to share their data with businesses or for commercial purposes. Many participants (N=6) in our study indicated they would only use data donation apps if they were assured their data would be used exclusively for research purposes. They stressed that their data should not be used for marketing or financial gain, but rather to further knowledge and provide new research perspectives. For instance, P8 said: *"I would share my data only when I'm sure that this data will be collected just for research and not for any other reason"*.

Privacy and Anonymity Guarantees Impact Sharing. Participants expressed their willingness to donate their data if the researchers possessed a high level of expertise in privacy and security, along with a commitment to anonymous data collection. Additionally, they emphasized that their confidence in sharing data would increase if the research institute operating the app were based in a country with strong data privacy laws and protections. Moreover, one participant (P2) emphasized the significance of protecting privacy by ensuring data collection from a large and diverse

population: *"If the study involves a small number of participants, I wouldn't agree to share my data. This is because I think that with a limited group of individuals, the risk of being identified increases. Thus, it is important for me that the study then has a diverse and large number of participants"*.

> **Key Findings:** In summary, participants highly prioritized the recipient of donated data, favoring research institutes over companies. They desired control over the collected data and its recipient but expressed concerns that exercising such control could introduce complexity on their end, and they may not be able to benefit from this option due to their lack of expertise. Additionally, they desired anonymity for their data while also wanting the ability to track their data within the system to monitor its use. However, these two conflicting desires may create tension, as once records are anonymized, they can no longer be linked to individuals, thus disabling control after anonymization.

4.3 Perceptions of Data Storage & Access

We explored participants' views on the storage location of collected data and their beliefs regarding entities that might have access to it.

Perceived Storage Locations Vary. The participants had various ideas regarding the storage location of the collected data. The most common expectation was that the data would be stored on a server. However, there were different views on where this server would be located. Most participants (N=14) believed that the server would be within the research institute that offers the app, such as P3, who said, *"It will be stored on a server on the researchers' side, and this server will be connected to a person's smartwatch"*. Some (N=3) mentioned that the server might be located within the software company responsible for the app's technical development. Other expectations included storage on a government server (N=1) or on cloud servers (N=5). One participant (P7) demonstrated advanced technical knowledge by proposing a distributed system, envisioning a network of interconnected servers where data could be stored. This participant also suggested that donated data could be divided across different parts of the network based on its nature, though not necessarily on all servers.

Six participants believed that data should only be stored on the user's smartphone or smart device (e.g., fitness tracker). Others (N=4) suggested a combination, with data stored on both the user's device and a server owned by the research institute. P1 and P11 proposed that data could be stored across all components and devices integrated into the infrastructure.

Two participants highlighted the importance of storing data in the same country where the users of the app live. Others (N=2) noted that the choice of storage location is influenced and governed by state data protection regulations and laws, e.g., P5 stated: *"If we're discussing this scenario in the context of Germany, I think that, in compliance with data protection laws, the data should be stored within the country"*.

Researchers Have Access, But Others Might Too. When the participants were asked specifically regarding their perceptions of who might have access to the donated raw data, we again found different perceptions. First, almost all participants identified researchers as the primary group with access. According to three participants, after the app has collected the data, those who donated it, the users, will undoubtedly still have access to it. Their rationale behind this was that since the users own the data they contribute, they inherently possess the right to access it at any time.

Perhaps interestingly, two participants (P16 & P19) held the perspective that data contributed through the app might be also accessible to official authorities, such as the health ministry or other health and social care agencies. They expected that these authorities could have an interest in examining the collected data to gain insights into the citizens.

Five participants (P7, P9, P11, P17 & P18) believed that individuals, such as app developers, system administrators, or service providers like internet service providers or cloud service providers might have the capability to access the data contributed through the app, e.g., P7 said: *"I guess there are always some kind of administration people who are not really interested in the data itself but in organizing all the structure of the network, and I guess they could also get some kind of access. But I guess they would not need to use the data for their job"*. Additionally, they emphasized that the companies responsible for manufacturing the smartwatches worn by users could also potentially access the users' data.

Finally, two participants mentioned that anyone or any entity within the infrastructure could potentially access the data. One of the two specified that this access should only be permitted with the user's consent.

Lack of Awareness About Access Control. Most participants were unaware of the possibility of having different levels and types of access to their data. Only three participants recognized the importance of access control and stressed that only individuals with proper permissions should be allowed to access the data. However, participants struggled describing who grants these permissions; some pointed to servers, developers, or researchers as responsible entities, whereas others argued that the users themselves are responsible. A sample quote by P11: *"I would say the person who gave the data, who is like the origin of the data, has the most right to determine who can get access to it"* or P10: *"I would like to think that only limited people have access to my data. And, it's important to clarify that granting access to an institution doesn't automatically grant access to all employees within that institution"*.

> **Key Findings:** In sum, the participants expressed a variety of possible locations where the data would be stored and expected that while researchers would have access to their donated data, other parties, including themselves, would also have access. Additionally, findings show that participants had limited knowledge about access control.

4.4 Perceptions of Privacy & Anonymity

Our aim was to acquire a deeper understanding of participants' speculative mental models regarding privacy in data donation apps. We initiated our exploration by questioning which data should be protected to maintain users' privacy. This also involved pinpointing potential threats or sources that this data should be protected from, determining the party accountable for ensuring this protection, and exploring different methods for achieving user privacy. After

gathering the participants' opinions on data privacy protection, we introduced the concept of anonymity and asked about their understanding of it.

Obvious PII is Considered Sensitive, Medical Data Not. All participants provided a range of examples of medical and demographic data that a medical data donation app could gather. Most examples of medical data provided by participants include information that can be collected by fitness trackers or wearable devices, such as heart rate, blood pressure, temperature, steps, and sleep patterns. They also mentioned self-reported data, including personal and family health history, symptoms, blood type, and current medications.

When participants were asked about the data they perceived as sensitive, the majority of them highlighted information related to PII and demographics, such as name (N=10) and address/location (N=9), with phone numbers coming closely after. Subsequent mentions included national or social IDs, gender, age, birth date, bank account details, occupation, religion, national or social identifiers, education level, phone number, race, and workplace location. Examples of participants' statements include P1 mentioning, *"I think it's probably the name and probably also the location because it's easier to identify a certain person from the location"*, and P14 stating, *"The address, the last name, and the phone number are the most important things to be protected. The other information is not that sensitive"*. All participants emphasized the necessity of protecting demographic data due to its capability of identifying users or disclosing their true identities, demonstrating a strong understanding of the sensitivity associated with demographics.

Only six participants recognized the sensitivity of medical data and the need for its protection. A sample comment is given by P22: *"The person's medical history or the medical history of the person's family should be protected because if this type of information is revealed, it can be used against the person to ruin his or her life, for example, the person may lose his or her work or reputation"*.

Two participants held the viewpoint that all data collected by the app from users was sensitive and should be protected, e.g., P18: *"I think all the data that is provided by the users to the app should be protected to ensure the unlinkability between people and their data"*.

One participant (P3) expressed a willingness to share all of their data with researchers, stating that they did not possess any specific information they deemed sensitive: *"There is no personal medical information about me that I consider myself sensitive. I'm not only talking about medical data but also personal information. Even my name, I don't think that is very sensitive for me"*.

Lack of Awareness About Metadata. We found that most participants displayed limited awareness regarding the collection and sensitivity of metadata. Two participants even believed their location-based data would not be gathered. For example, P1 said, *"I think they will respect my privacy, so they will not ask for too much demographic or collect location-based data"*.

Only P2 and P5 mentioned examples of sensitive metadata and recognized that it could be used to link app users to their donated data, e.g., P5: *"The app could collect some metadata that can be traced back, such as the server storing the path from which the data originates. You can then trace the specific points where the data has traveled and ultimately trace it back to the person"*. Examples of metadata provided by these two participants included location metadata (e.g., IP addresses), device metadata (e.g., MAC addresses), the duration taken to complete the questionnaire, and the time when the questionnaire was completed. These participants emphasized the importance of protecting metadata, with P2 suggesting that if metadata is absolutely necessary, it should be stored separately and deleted after a certain period.

Researchers Lead Protection, with Room for Shared Responsibility. When we asked about who is responsible for privacy and anonymity protection, most of the participants (N=17) pointed to the research institute that provides the app as the main entity responsible for leading data protection efforts. However, interestingly, one participant (P5) believed that anonymity preservation should not be the responsibility of researchers but rather of an external party: *"Anonymity should not be done by researchers because I think that's a conflict of interest. You need an external company to maintain anonymity software running in the cloud or on a server. If it's open source, well, people can check the code, but still, someone has to still maintain it. You could also have a new company every five years or something. Like to switch out so you don't have the same partner for a long time. That's what I'd recommend"*.

Many participants (N=9) emphasized that technical companies involved in developing data donation apps or manufacturing smart devices used by users to generate medical data should also oversee data protection. Some participants (N=5) stated that the responsibility should also lie with the government by implementing laws and regulations to ensure user protection. One participant (P24) mentioned that users should protect their data when donating it to the app by using security tools (e.g., antivirus software) on their devices. Only one participant (P12) mentioned that users are responsible for preserving their privacy by choosing what to disclose: *"The users themselves just have to look at what kind of data is collected and think about, okay, could I possibly imagine any of these data to identify myself? If I were given this data, could I identify someone with it? And then, if it looks good, the user can participate"*.

Perceived Threats. We asked participants about the potential threats that data should be protected against. Table 2 provides a summary of the threats they mentioned. When we explored who might have the potential to violate user privacy or break their anonymity, the responses of participants included either researchers or an external attacker who gains control over users' devices or servers. For example, P5 said: *"I'm assuming the institution or the group responsible for creating the system is a good actor. The bad actors who want to compromise anonymity are coming from the outside"*.

Interdependent Privacy Overlooked. Only one participant (P8) raised concerns about the privacy implications of gathering sensitive data on individuals who are not app users themselves, such as the friends and family of the app user. P8 argued that while information on family health histories, like parental cancer risks, could be valuable for understanding health conditions, collecting such data is problematic and raises serious privacy and ethical issues, particularly because it involves individuals who are not using the app and have not given explicit consent for their data to be collected.

Threat	Description
Unauthorized Access & Data Breaches	Concerns about donated data being stolen or accessed by those who do not have permission.
Data Misuse & Discrimination	Concerns about unethical use of donated data and the potential negative consequences. For example, P22 was concerned that researchers might disclose sensitive information about a donor, especially regarding stigmatized conditions, potentially damaging the donor's reputation and leading to discrimination. P16 feared that researchers might reveal the donated data to insurance companies which could lead to higher premiums.
Phishing Attacks	Concerns about attempts to obtain sensitive data through fake data donation apps, with P5 highlighting the need for protection against such attacks.

Table 2: Perceived Threats

Anonymity as Prerequisite. The majority of participants demonstrated a general familiarity with anonymity. For instance, 20 out of 24 confirmed they had heard the anonymity term before. Most participants defined anonymity as data that couldn't be traced back to the individual who provided it. Also, they understood the implications of breaking anonymity. For example, P1 mentioned, *"That the health data is connected to the person, the name, the birth date or maybe also the location"*. Similarly, P21 stated, *"It means discovering the identity of the person who gave data, which means he no longer has privacy"*.

We asked participants whether they would be willing to use a medical data donation app if they were aware that this app could link their donated data to their name, mobile phone number, or location. Fourteen participants completely declined to donate their data under such conditions. For instance, P2 said: *"If I knew that the data could be linked or thought that the data could be linked back, this is the point where I would say no so that I wouldn't participate"*. Also, P23 stated: *"No, because it is very difficult to trust researchers in this case"*. Only four participants were open to donate their data even if anonymity was not assured. Six participants expressed that ensuring anonymous data donation is very important to them, but they might agree under certain circumstances to denote their data when anonymity is not ensured. These conditions include perceiving the donated data as non-sensitive, having trust in the research institute's commitment to user protection, and believing that the country in which the research institute is located would enforce privacy protection. A sample comment by P10: *"I would be more specific about which data to donate. So for example I know there are medical information or medical data that I would not mind being linked to me personally because they are maybe more common. For example headaches, flu, and some illnesses that you have that everyone has. But as soon as it comes to very specific things like very specific illnesses or very specific cases then I would like to keep that to myself. If I can be linked to these, that could be used against me in some way"*.

Perceived Privacy & Anonymity Preservation Methods. When we asked participants how medical data donation apps could maintain privacy, we received a wide range of responses. Participants described various methods that align with common protection techniques, although most were not familiar with the specific names of techniques. The mentioned ones included encryption (N=4), data aggregation (N=4), access control (N=3), anonymity (N=3), and pseudonymization (N=2). One participant stressed the importance of raising awareness, suggesting that countries should educate their residents about data significance, handling, and self-protection. Yet, another participant emphasized data protection can be achieved through state laws and official data protection regulations. One participant proposed protecting data by making users donate only outdated data, as according to their understanding, this data would no longer relate to the same individual; P11: *"If they only have data from the past, it's not actually about me. It's about past me that's quite different from me now"*.

When we asked specifically about how the anonymity of users can be ensured in medical data donation apps, some participants (N=5) mentioned traditional security techniques like encryption. Other participants (N=7) suggested that the app should obscure or eliminate PII from the donated data, while two participants stated that no PII should be collected at all. Additionally, some participants (N=2) proposed that ensuring anonymity could involve deliberately supplying inaccurate data to researchers. For example, P14 said, *"A person can give wrong or fake answers like saying he is female although he is male. But this sure will affect the research results"*.

Furthermore, some participants described methods that align with the following techniques: pseudonymization (N=3), data aggregation (N=2), suppression (N=1), and generalization (N=1). For instance, P6 referred to generalization, stating, *"Well, I know some techniques to anonymize data. For example, if the user inserted the exact age, like 28, then it should be converted to an age range. So we will end up with 20 to 30 or 25 to 30, things like that"*.

Data shuffling (N=2) was also brought up; e.g., P2 mentioned, *"I hope they also change the order in which the data was collected. So you cannot say, okay, this particular data came at this particular point in time, or it came from this location"*. Another participant held the perspective that anonymity could be established by opting for data collection through paper-based questionnaires rather than relying on apps or fitness trackers.

Privacy & Anonymity Awareness Issues. Many participants exhibited limited or inaccurate knowledge regarding the protection measures, such as P9: *'I don't know how the data could be protected because actually, I don't know how the data safety in Germany or Europe works. I heard of that already often, but I don't know. Especially about the medical data, I don't have any idea"*. Also, several had awareness issues about how their privacy or anonymity could be compromised and who might be motivated to do so. For example,

nine participants believed that breaking user anonymity was not feasible as long as the data they provided did not contain explicit personal identifiers, which was proven wrong in [61]. E.g., P14: *As long as the data does not include a last name, email address, or location, no one could know the person's identity or link data to him or her".* or P18: *If the data is just medical data and no birthdate or birthplace, I think it is very difficult to compromise anonymity in this case".*

Some participants (N=3) thought that the demographic information could not be used to break the anonymity of individuals who provided their data via a data donation app. This was rooted in a missing distinction between the app's user base and the larger population in a nation or worldwide: *'Because there are millions of people in the world who have the same age, weight, height, and other characteristics"* (P23). Also, the sensitivity of other data types, such as donated medical data, was rarely mentioned even though it is possible to identify individuals based on such data [48].

Only three participants (P2, P7 & P21) recognized the importance of unique demographics, medical data, or distinct data patterns in comparison to other app users as potential factors contributing to de-anonymization. E.g., P2: *'But if I know that the study is not very large, then my nationality or my race could be traced back to me. I previously took part in a study where I was the only participant of my nationality. I had concerns that if they included a statement from a participant of my nationality in their report, it could easily be traced back to me".*

Generally, we found that the participants who had knowledge gaps regarding privacy and anonymity tended to be more resistant to considering the use of medical data donation apps.

> **Key Findings:** The participants highly valued anonymity. They did not want to be identified in any way through the shared data. While they recognized that obvious data, like demographics and PII, could identify them, they did not consider that medical data (e.g., a specific disease) might also reveal their identity. Additionally, participants feared discrimination as a potential negative consequence of de-anonymization.

4.5 Comparison to Existing Apps

Most participants, across all types of speculative mental models, perceived that the donated data is sent directly from users to researchers or research institutes, aligning with the infrastructure of CDA. An exception was participant P5, who anticipated the presence of an anonymizer entity between users and the research institute, similar to SafeVac. However, this participant had higher expectations regarding the role of the anonymity entity. They anticipated the anonymizer not only routing data to the research institute, as in SafeVac, but also anonymizing the data before transmission: *"Here, we will basically have a sort of anonymizer. So all the data goes first to it to be anonymized, then the anonymizer directly sends the data in the anonymized format to the institute".*

Pseudonymization, as the approach used in CDA and SafeVac to protect users' privacy, matched the protection method expected by three participants (note that they did not name the approach but instead described what aligns with how it works). However, the majority of participants, including these three, strongly expressed a desire for anonymity when donating their medical data via apps. They wanted their data to be untraceable to them or their locations. Pseudonymization alone cannot ensure this level of protection [26, 61]. This suggests that CDA and SafeVac do not provide the protection guarantees that participants need. Interestingly, many participants, including P2 & P12, who claimed familiarity with the apps, anticipated that the apps were employing much stronger measures, such as data removal, shuffling, aggregation, and generalization. For instance, P2 believed that data would be anonymized locally, possibly aggregated with data from other users before being received by the research institute. Furthermore, some participants (N=3) expected the apps to implement access control measures to restrict access to the donated data and allow only authorized individuals to have access. However, neither CDA nor SafeVac provided any information about whether they implement such measures.

While most participants drew infrastructures close to that of CDA, very few considered the app's ability in this case to link users to their data through the IP address. Given that only four participants agreed to donate their data if the app could link it to their location/address, it suggests a disparity between participants' understanding of privacy and anonymity within the data donation apps and their actual privacy and anonymity needs. Moreover, this highlights a gap between the functionalities of existing apps and participants' preferences, as most participants want an app that ensures strong anonymity, including hiding the origin of the data, i.e., concealing location or IP address.

5 Discussion

Our findings from interviewing participants suggest the importance of user-friendly data donation app designs to encourage people to use these apps. While privacy was considered in many debates on data donation apps (cf. [64]), our study focus was more on anonymity as it is crucial in the domain of data donation where donors have to be certain that the sensitive data that could be linked to them, is correctly anonymized.

Perspective of Medical Researchers. In addition to investigating the perspectives of potential future users of medical data donation apps, we also explored the viewpoints of medical researchers. Before beginning our study, we held discussions with members of our medical faculty and leading researchers from the Paul Ehrlich Institute, which provides the SafeVac app. From these meetings, it was clear that medical data donation is highly valued by researchers. While there was strong support for protecting donor privacy and complying with regulations like GDPR, researchers also expressed a need for data to be flexible enough for various types of analysis. They discussed the trade-off between privacy and utility, expressing concerns that excessive anonymization might reduce the data's usefulness for research. For instance, they mentioned that anonymizing data might involve removing outliers to mitigate re-identification risks, but these outliers can sometimes be crucial for analyses. They emphasized the importance of anonymizing data in a way that preserves its utility. Additionally, researchers from the Paul Ehrlich Institute highlighted the need to connect data points that come from the same donor, even if the data is anonymized and the researchers do not know the donor's real identity.

Recommendations. Based on our findings and collected insights, we discuss key design recommendations for creating a user-centric medical data donation app:

1) Make data donation research exclusive. Our participants have expressed that data donation apps should be exclusively for research purposes and have indicated that they would refuse to share their medical data if it could be used for commercial gain. Based on that, we recommend that the data donation apps should not be driven by any profit motives and limited to research only or the participants can opt-out of commercial studies by companies.
2) Make data recipients, studies & results transparent. When it comes to collecting medical data, transparency is crucial for establishing user trust. Similar to other privacy-sensitive domains, medical data donation apps should clearly outline what information they gather and why. Additionally, they should explain the types of studies that may use the collected data, how the data is stored, who can access it, and when it will be deleted. As stated above, users do not receive a direct personal benefit, yet might be driven by the benefit of society as a whole. Based on that, we recommend notifying app users about research results, new treatments, or similar where their data was used. This could make users proud of data donation yet needs further investigation in future work.
3) Make sharing highly customizable. Even though privacy and anonymity have different levels of control, we argue that control should not be completely taken away from individuals. Some of our participants were only willing to donate specific data or wanted to select which data about them to donate. This aligns with findings from other privacy-sensitive domains, such as IoT [40, 62], and studies on medical data sharing [13, 67]. Instead of providing only all-or-nothing settings, the app should enable users to easily and conveniently choose which data to donate and specify which studies can use it.
4) Data minimization. Some participants were concerned about sharing information beyond what was necessary or relevant to the research. They worried that such data might not be used for research purposes or could be misused. Hence, we recommend that data donation apps refrain from gathering any non-essential demographic or medical information from their users, and not collecting any data that could explicitly identify an individual. We observed strong rejection among participants regarding sharing their addresses or locations. Hence, we highly advise against collecting this kind of information. However, we are aware that explorative research endeavors might collect data that later on proves to be not useful. Such cases must be clearly communicated to their users.
5) Ensure anonymity by default. Some participants would donate any kind of data if they were assured that the app maintains unlinkability between users and their donated data. Our participants valued their anonymity and were hesitant to donate if their information could be traced back to them. However, they also expressed difficulty in judging what data can be used to track them. Therefore, to gain user trust and willingness to share data, we recommend that apps deploy strong anonymity measures by default that safeguard against de-anonymization risks in both data and communication. The users should be taken out of the loop by allowing to only donate anonymous data. Moreover, medical data donation apps should clearly communicate the level of security they provide in a way that is easy for the average user to understand. Tracing apps in the COVID-19 pandemic showed that this is not an easy task [64]. Several countries used different app infrastructures offering various privacy levels. Further, it might be possible to de-anonymize datasets in the future with novel algorithms. Future work should investigate techniques for a) robust anonymity that lasts long-term and b) means communicating this to users in a verifiable way allowing users to verify the anonymization of their donated data.
6) Balancing Privacy and Data Utility. All the medical researchers we spoke with emphasized the need to balance privacy protection with the requirement for high-quality, useful data that can support a variety of research analyses. As known, anonymization techniques vary in their utility and privacy capabilities, and no single technique is universally applicable. Therefore, before designing a data donation app, we recommend consulting with potential data recipients (researchers who will use the data) to identify scenarios where the data might be anonymized in a way that renders it not useful for their analyses. This will help in selecting the most appropriate anonymization technique that maximizes privacy while meeting researchers' utility requirements.

Future Work. In future research, it would be interesting to explore how privacy mental models differ between users and non-users of medical data donation apps. Also, it would be worthwhile to investigate the differences between the privacy mental models of participants from different cultures, as the cultural factor has been shown by many studies to have a significant impact on participants' perceptions and awareness of privacy.

6 Conclusion

This paper examines the perceptions and expectations of non-users of medical data donation apps. Our findings reveal that participants had difficulty understanding how these apps work, highlighting the need for clearer information. Trust, transparency, strong security, and full anonymity were essential for their participation. Although participants understood what data could be collected, they lacked awareness about data sensitivity and protection methods, including anonymity. Privacy concerns, such as data breaches and discrimination, were noted, but understanding of these risks was limited. Those with less knowledge about privacy protections were less willing to donate data. We compared participants' expectations with two existing apps and identified gaps between the apps' protections and user needs, offering design recommendations to better align with privacy and anonymity expectations.

Acknowledgments

This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA - 390781972.

References

[1] 2020. COVID-19: Implications for the EU and its economy. https://www.europarl.europa.eu/RegData/etudes/BRIE/2020/652711/IPOL_BRI(2020)652711_EN.pdf.
[2] 2020. SafeVac FAQ. https://www.pei.de/DE/service/faq/coronavirus/faq-coronavirus-safevac-app-tabelle.html.
[3] Noura Abdi, Kopo M. Ramokapane, and Jose M. Such. 2019. More than Smart Speakers: Security and Privacy Perceptions of Smart Home Personal Assistants. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS '19)*. USENIX Association, Berkeley, CA, USA, 1–16.

[4] Mhairi Aitken, Jenna de St. Jorre, Claudia Pagliari, Ruth Jepson, and Sarah Cunningham-Burley. 2016. Public responses to the sharing and linkage of health data for research purposes: a systematic review and thematic synthesis of qualitative studies. *BMC medical ethics* 17 (2016), 1–24.

[5] Abdulmajeed Alqhatani and Heather Richter Lipford. 2019. " There is nothing that I need to keep secret": Sharing Practices and Concerns of Wearable Fitness Data.. In *SOUPS@ USENIX Security Symposium*.

[6] Norm Archer and Mihail Cocosila. 2014. Canadian patient perceptions of electronic personal health records: An empirical investigation. *Communications of the Association for Information Systems* 34, 1 (2014), 20.

[7] Khadija Baig, Reham Mohamed, Anna-Lena Theus, and Sonia Chiasson. 2020. " I'm hoping they're an ethical company that won't do anything that I'll regret" Users Perceptions of At-home DNA Testing Companies. In *Proceedings of the 2020 CHI conference on human factors in computing systems*. 1–13.

[8] Rahime Belen-Saglam, Jason RC Nurse, and Duncan Hodges. 2022. An investigation into the sensitivity of personal information and implications for disclosure: A UK perspective. *Frontiers in Computer Science* 4 (2022), 908245.

[9] Matthew Bietz, Kevin Patrick, and Cinnamon Bloss. 2019. Data donation as a model for citizen science health research. *Citizen Science: Theory and Practice* 4, 1 (2019).

[10] Ted Boren and Judith Ramey. 2000. Thinking aloud: Reconciling theory and practice. *IEEE transactions on professional communication* 43, 3 (2000), 261–278.

[11] Christine L Borgman. 1999. The user's mental model of an information retrieval system: an experiment on a prototype online catalog. *International journal of human-computer studies* 51, 2 (1999), 435–452.

[12] Virginia Braun and Victoria Clarke. 2012. *Thematic analysis*. American Psychological Association.

[13] Richard Brown, Elizabeth Sillence, Lynne Coventry, Emma Simpson, Jo Gibbs, Shema Tariq, Abigail C. Durrant, and Karen Lloyd. 2022. Understanding the attitudes and experiences of people living with potentially stigmatised long-term health conditions with respect to collecting and sharing health and lifestyle data. *Digital health* 8 (2022), 20552076221089798.

[14] Lorina Buhr, Silke Schicktanz, Eike Nordmeyer, et al. 2022. Attitudes toward mobile apps for pandemic research among smartphone users in Germany: national survey. *JMIR mHealth and uHealth* 10, 1 (2022), e31857.

[15] Tânia Carvalho, Nuno Moniz, Pedro Faria, and Luís Antunes. 2023. Survey on Privacy-Preserving Techniques for Microdata Publication. *ACM Comput. Surv.* 55, 14s, Article 309 (jul 2023), 42 pages.

[16] Victoria Clarke and Virginia Braun. 2013. Successful qualitative research: A practical guide for beginners. *Successful qualitative research* (2013), 1–400.

[17] D. Mentzer D. Oberle and G. Weber. 2020. Befragung zur Verträglichkeit der Impfstoffe gegen das neue Coronavirus (SARS-CoV-2) mittels Smartphone-App SafeVac 2.0. https://www.pei.de/SharedDocs/Downloads/EN/newsroom-en/pharmacovigilance-bulletin/single-articles/2020-safevac-app-en.pdf?__blob=publicationFile&v=3.

[18] Daniel Diethei, Jasmin Niess, Carolin Stellmacher, Evropi Stefanidi, and Johannes Schöning. 2021. Sharing heartbeats: motivations of citizen scientists in times of crises. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.

[19] Tamara Dinev and Paul Hart. 2006. An extended privacy calculus model for e-commerce transactions. *Information systems research* 17, 1 (2006), 61–80.

[20] European Union. 2011. General Data Protection Regulation (GDPR) - Recital 35. https://gdpr-info.eu/recitals/no-35/.

[21] Roy Evans and Eamonn Ferguson. 2014. Defining and measuring blood donor altruism: a theoretical approach from biology, economics and psychology. *Vox sanguinis* 106, 2 (2014), 118–126.

[22] Keolu Fox. 2020. The illusion of inclusion—The "All of Us" research program and indigenous peoples' DNA. *New England Journal of Medicine* 383, 5 (2020), 411–413.

[23] Thomas Franke, Christiane Attig, and Daniel Wessel. 2019. A personal resource for technology interaction: development and validation of the affinity for technology interaction (ATI) scale. *International Journal of Human–Computer Interaction* 35, 6 (2019), 456–467.

[24] Benjamin CM Fung, Ke Wang, Rui Chen, and Philip S Yu. 2010. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (Csur)* 42, 4 (2010), 1–53.

[25] Marco Furini, Silvia Mirri, Manuela Montangero, and Catia Prandi. 2020. Can IoT wearable devices feed frugal innovation?. In *Proceedings of the 1st Workshop on Experiences with the Design and Implementation of Frugal Smart Objects*. 1–6.

[26] Sarah Abdelwahab Gaballah, Lamya Abdullah, Mina Alishahi, Thanh Hoang Long Nguyen, Ephraim Zimmer, Max Mühlhäuser, and Karola Marky. 2024. Anonify: Decentralized Dual-level Anonymity for Medical Data Donation. *Proceedings on Privacy Enhancing Technologies* 3 (2024), 1–15.

[27] Nanibaa'A Garrison, Nila A Sathe, Armand H Matheny Antommaria, Ingrid A Holm, Saskia C Sanderson, Maureen E Smith, Melissa L McPheeters, and Ellen W Clayton. 2016. A systematic literature review of individuals' perspectives on broad consent and data sharing in the United States. *Genetics in Medicine* 18, 7 (2016), 663–671.

[28] Kit Huckvale, John Torous, and Mark E Larsen. 2019. Assessment of the data sharing and privacy practices of smartphone apps for depression and smoking cessation. *JAMA network open* 2, 4 (2019), e192542–e192542.

[29] Maximilian Häring, Eva Gerlitz, Christian Tiefenau, Matthew Smith, Dominik Wermke, Sascha Fahl, and Yasemin Acar. 2021. Never ever or no matter what: Investigating Adoption Intentions and Misconceptions about the Corona-Warn-App in Germany. In *Seventeenth Symposium on Usable Privacy and Security, SOUPS 2021, August 8-10, 2021*. USENIX Association, 77–98. https://www.usenix.org/conference/soups2021/presentation/acar

[30] Philip Nicholas Johnson-Laird. 1983. *Mental models: Towards a cognitive science of language, inference, and consciousness*. Number 6. Harvard University Press.

[31] David Jonassen and Young Hoan Cho. 2008. Externalizing mental models with mindtools. *Understanding models for learning and instruction* (2008), 145–159.

[32] Bailey Kacsmar, Kyle Tilbury, Miti Mazmudar, and Florian Kerschbaum. 2022. Caring about Sharing: User Perceptions of Multiparty Data Sharing. In *31st USENIX Security Symposium (USENIX Security 22)*. 899–916.

[33] Ruogu Kang, Laura Dabbish, Nathaniel Fruchter, and Sara Kiesler. 2015. {"My} Data Just Goes {Everywhere:"} User Mental Models of the Internet and Implications for Privacy and Security. In *Eleventh symposium on usable privacy and security (SOUPS 2015)*. 39–52.

[34] Florian Kohlmayer, Ronald Lautenschläger, and Fabian Prasser. 2019. Pseudonymization for research data collection: is the juice worth the squeeze? *BMC medical informatics and decision making* 19 (2019), 1–7.

[35] Patrick Kühtreiber, Viktoriya Pak, and Delphine Reinhardt. 2022. Replication: The Effect of Differential Privacy Communication on German Users' Comprehension and Data Sharing Attitudes. In *Eighteenth Symposium on Usable Privacy and Security (SOUPS 2022)*. 117–134.

[36] Todd Kulesza, Simone Stumpf, Margaret Burnett, Sherry Yang, Irwin Kwan, and Weng-Keen Wong. 2013. Too much, too little, or just right? Ways explanations impact end users' mental models. In *2013 IEEE Symposium on visual languages and human centric computing*. IEEE, 3–10.

[37] Hyunsoo Lee, Soowon Kang, and Uichin Lee. 2022. Understanding privacy risks and perceived benefits in open dataset collection for mobile affective computing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 2 (2022), 1–26.

[38] Naresh K Malhotra, Sung S Kim, and James Agarwal. 2004. Internet users' information privacy concerns (IUIPC): The construct, the scale, and a causal model. *Information systems research* 15, 4 (2004), 336–355.

[39] Karola Marky, Sarah Prange, Max Mühlhäuser, and Florian Alt. 2021. Roles matter! Understanding differences in the privacy mental models of smart home visitors and residents. In *Proceedings of the 20th International Conference on Mobile and Ubiquitous Multimedia*. 108–122.

[40] Karola Marky, Alexandra Voit, Alina Stöver, Kai Kunze, Svenja Schröder, and Max Mühlhäuser. 2020. "I Don't Know How to Protect Myself": Understanding Privacy Perceptions Resulting from the Presence of Bystanders in Smart Environments. In *Proceedings of the Nordic Conference on Human-Computer Interaction* (Tallinn, Estonia). ACM, New York, NY, USA, Article 4, 11 pages. https://doi.org/10.1145/3419249.3420164

[41] Patrick Jäger Martin Tschirsich and André Zilch. 2020. Blackbox-Sicherheitsbetrachtung Corona-Datenspende-App des RKI. https://www.ccc.de/system/uploads/297/original/CCC_Analyse_Datenspende.pdf.

[42] Gary T Marx. 1999. What's in a Name? Some Reflections on the Sociology of Anonymity. *The information society* 15, 2 (1999), 99–112.

[43] Nora McDonald and Andrea Forte. 2020. The politics of privacy theories: Moving from norms to vulnerabilities. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–14.

[44] Nora McDonald, Rachel Greenstadt, and Andrea Forte. 2023. Intersectional thinking about PETs: A study of library privacy. *Proceedings on Privacy Enhancing Technologies* (2023).

[45] Roisin McNaney, Catherine Morgan, Pranav Kulkarni, Julio Vega, Farnoosh Heidarivincheh, Ryan McConville, Alan Whone, Mickey Kim, Reuben Kirkham, and Ian Craddock. 2022. Exploring Perceptions of Cross-Sectoral Data Sharing with People with Parkinson's. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–14.

[46] Donald A Norman. 2014. Some observations on mental models. In *Mental models*. Psychology Press, 15–22.

[47] Briony J Oates, Marie Griffiths, and Rachel McLean. 2022. *Researching information systems and computing*. Sage.

[48] Iyiola E Olatunji, Jens Rauch, Matthias Katzensteiner, and Megha Khosla. 2022. A review of anonymization for healthcare data. *Big data* (2022).

[49] Rebecca Panskus, Max Ninow, Sascha Fahl, and Karola Marky. 2023. Privacy Mental Models of Electronic Health Records: A German Case Study. In *Nineteenth Symposium on Usable Privacy and Security (SOUPS 2023)*. 525–542.

[50] David J Phillips. 2004. Privacy policy and PETs: The influence of policy regimes on the development and social implications of privacy enhancing technologies. *New Media & Society* 6, 6 (2004), 691–706.

[51] Gesine Richter, Christoph Borzikowsky, Bimba Franziska Hoyer, Matthias Laudes, and Michael Krawczak. 2021. Secondary research use of personal medical data:

patient attitudes towards data donation. *BMC medical ethics* 22, 1 (2021), 1–10.
[52] RKI. 2019. Corona Data Donation Project. https://corona-datenspende.de/science/en/.
[53] Mike Scaife and Yvonne Rogers. 1996. External cognition: how do graphical representations work? *International journal of human-computer studies* 45, 2 (1996), 185–213.
[54] Stefan Schneegass, Romina Poguntke, and Tonja Machulla. 2019. Understanding the impact of information representation on willingness to share information. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–6.
[55] Eva-Maria Schomakers, Chantal Lidynia, and Martina Ziefle. 2020. All of me? Users' preferences for privacy-preserving data markets and the importance of anonymity. *Electronic Markets* 30, 3 (2020), 649–665.
[56] Annett Schwamborn, Richard E Mayer, Hubertina Thillmann, Claudia Leopold, and Detlev Leutner. 2010. Drawing as a generative activity and drawing as a prognostic activity. *Journal of Educational Psychology* 102, 4 (2010), 872.
[57] Christina Schön, Roland Speidel, and Sabine Welsch. 2023. Minimum Wage and Mini-Job Threshold will rise on January 1, 2024. https://www.bdo.de/en-gb/insights/updates/tax-legal/minimum-wage-and-mini-job-threshold
[58] Emily Seltzer, Jesse Goldshear, Sharath Chandra Guntuku, Dave Grande, David A Asch, Elissa V Klinger, and Raina M Merchant. 2019. Patients' willingness to share digital health and non-health data for research: a cross-sectional study. *BMC medical informatics and decision making* 19 (2019), 1–8.
[59] Anya Skatova and James Goulding. 2019. Psychology of personal data donation. *PloS one* 14, 11 (2019), e0224240.
[60] Joanna Sleigh. 2018. Experiences of donating personal data to mental health research: an explorative anthropological study. *Biomedical Informatics Insights* 10 (2018), 1178222618785131.
[61] Latanya Sweeney. 2002. k-anonymity: A model for protecting privacy. *International journal of uncertainty, fuzziness and knowledge-based systems* 10, 05 (2002), 557–570.
[62] Madiha Tabassum, Tomasz Kosinski, and Heather Richter Lipford. 2019. " I don't own the data": End User Perceptions of Smart Home Device Data Practices and Risks. In *Fifteenth symposium on usable privacy and security (SOUPS 2019)*. 435–450.
[63] Nora Tophof and Maximilian Tischer. 2024. Data Donation: Better Health and Quality of Life for All. https://www.data4life.care/en/library/journal/data-donation-in-medicine/.
[64] Christine Utz, Steffen Becker, Theodor Schnitzler, Florian M Farke, Franziska Herbert, Leonie Schaewitz, Martin Degeling, and Markus Dürmuth. 2021. Apps against the spread: Privacy implications and user acceptance of COVID-19-related smartphone apps on three continents. In *Proceedings of the 2021 chi conference on human factors in computing systems*. 1–22.
[65] André Calero Valdez and Martina Ziefle. 2019. The users' perspective on the privacy-utility trade-offs in health recommender systems. *International Journal of Human-Computer Studies* 121 (2019), 108–121.
[66] Lev Velykoivanenko, Kavous Salehzadeh Niksirat, Noé Zufferey, Mathias Humbert, Kévin Huguenin, and Mauro Cherubini. 2021. Are those steps worth your privacy? Fitness-tracker users' perceptions of privacy and utility. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 4 (2021), 1–41.
[67] Torsten H Voigt, Verena Holtz, Emilia Niemiec, Heidi C Howard, Anna Middleton, and Barbara Prainsack. 2020. Willingness to donate genomic and other medical data: results from Germany. *European Journal of Human Genetics* 28, 8 (2020), 1000–1009.
[68] Rick Wash. 2010. Folk models of home computer security. In *Proceedings of the Sixth Symposium on Usable Privacy and Security*. 1–16.
[69] James Q Whitman. 2003. The two western cultures of privacy: Dignity versus liberty. *Yale LJ* 113 (2003), 1151.
[70] Yaxing Yao, Davide Lo Re, and Yang Wang. 2017. Folk models of online behavioral advertising. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. 1957–1969.
[71] Eric Zeng, Shrirang Mare, and Franziska Roesner. 2017. End user security and privacy concerns with smart homes. In *thirteenth symposium on usable privacy and security (SOUPS 2017)*. 65–80.
[72] Verena Zimmermann, Merve Bennighof, Miriam Edel, Oliver Hofmann, Judith Jung, and Melina von Wick. 2018. 'Home, smart home'–exploring End users' mental models of smart homes. *Mensch und Computer 2018-Workshopband* (2018).
[73] Verena Zimmermann, Paul Gerber, Karola Marky, Leon Böck, and Florian Kirchbuchner. 2019. Assessing Users' Privacy and Security Concerns of Smart Home Technologies. *i-com* 18, 3 (2019), 197–216. https://doi.org/10.1515/icom-2019-0015
[74] Noé Zufferey, Kavous Salehzadeh Niksirat, Mathias Humbert, and Kévin Huguenin. 2023. "Revoked just now!" Users' Behaviors toward Fitness-Data Sharing with Third-Party Applications. *Proceedings on Privacy Enhancing Technologies* 1 (2023), 1–21.

A Appendix

A.1 Codebook

The bullet points represent the categories of our coding tree. The frequency is given in brackets.

- **Motivation**
  - helping humanity (N=5)
  - improving research (N=10)
  - when data really needed (N=3)
  - no motivation (N=2)
  - getting a reward (N=3)
- **Expectations**
  - having control (N=10)
  - protecting data (N=3)
  - trust & reputation (N=13)
  - transparency (N=15)
  - data collected only for research (N=6)
  - diverse & large population (N=1)
- **Data Collection**
  - collected data (N=24)
  - not collected data (N=1)
  - sensitive data (N=22)
  - non-sensitive data (N=2)
- **Data Storage**
  - server (N=15)
  - user side (N=6)
  - research institute (N=14)
  - cloud (N=5)
  - database (N=1)
  - depending on laws & regulations (N=2)
  - everywhere in the infrastructure (N=2)
  - no idea (N=1)
- **Data Access**
  - access control (N=4)
  - data accessed by users (N=3)
  - data accessed by researchers (N=20)
  - data accessed by state (N=2)
  - data accessed by technicians (N=5)
  - data accessed by anyone (N=2)
- **Data Protection**
  - examples data needs protection (N=24)
  - protection by users (N=1)
  - protection by researchers (N=17)
  - protection by state (N=5)
  - protection by technicians (N=9)
  - protection by everyone (N=1)
  - protection against data breaches (N=3)
  - protection against data misuse (N=4)
  - protection against discrimination (N=3)
  - protection against phishing attacks (N=1)
  - protection using encryption (N=4)
  - protection using aggregation (N=4)
  - protection using anonymity (N=3)
  - protection using laws (N=1)
  - protection using awareness (N=1)
  - protection using access control (N=3)
  - protection using pseudonymization (N=2)
  - protection using a donation of old data (N=1)
- **Anonymity**
  - anonymization definition (N=20)
  - anonymity using pseudonymization (N=3)
  - anonymity using encryption (N=5)
  - anonymity using laws & consents (N=3)
  - anonymity using shuffling (N=2)
  - anonymity using aggregation (N=2)
  - anonymity using generalization (N=1)
  - anonymity using suppression (N=1)
  - anonymity using no PII collection (N=2)
  - anonymity using paper-based donation (N=1)
  - anonymity using data removal (N=7)
  - anonymity using incorrect data donation (N=2)
  - anonymity by research institute (N=3)
  - anonymity by an external company (N=3)
- **De-anonymization**
  - de-anonymization definition (N=22)
  - de-anonymization by an external attacker (N=2)
  - de-anonymization by researchers (N=2)
  - de-anonymization using location (N=5)
  - de-anonymization using demographics (N=4)
  - de-anonymization using medical data (N=1)
  - de-anonymization possible without PII in data (N=15)
  - de-anonymization difficult without PII in data (N=9)
  - accept non-anonymous donation (N=4)
  - refuse non-anonymous donation (N=14)
  - conditional accept non-anonymous donation (N=6)

ID	Familiarity with the anonymity term	Familiarity with data donation term	Experience with medical data donation	Usage Experience with data donation apps
P1	No	Yes	Yes (online)	No, but mentioned having prior information about them
P2	Yes	Yes	Yes (offline)	No
P3	Yes	No	Yes (offline)	No, but mentioned having prior information about them
P4	Yes	No	No	No
P5	Yes	Yes	No	No
P6	Yes	Yes	Yes (offline)	No
P7	Yes	No	No	No
P8	No	No	No	No
P9	Yes	No	No	No
P10	Yes	No	No	No
P11	Yes	Yes	Yes (offline)	No
P12	Yes	Yes	Yes (offline)	No, but mentioned having prior information about them
P13	Yes	No	Yes (offline)	No
P14	Yes	No	No	No
P15	No	No	No	No
P16	Yes	No	No	No
P17	Yes	No	No	No
P18	Yes	No	No	No
P19	Yes	No	No	No
P20	Yes	No	No	No
P21	Yes	No	No	No
P22	Yes	No	No	No
P23	Yes	No	No	No
P24	No	No	No	No

Table 3: The participants' familiarity with anonymity and data donation concepts, as well as their prior experience in data donation and its apps.

A.2 Details about Participants

Table 3 offers insights into participants' familiarity with anonymity and data donation concepts, along with their previous experiences related to medical data donation and associated applications.

A.3 Interview Questions

- *Have you ever heard about data donation?* If yes: *In which context?*
  - Let me explain to you what data donation in the scope of health data is: Data donation is a concept that aims to improve scientific research by giving citizens the opportunity to provide data concerning their health to researchers.
- *Have you ever donated your medical data, e.g., by participating in a questionnaire?*
  - If yes: *which context? What was the topic of the study? How was the data donated: paper-based or online? What types of data have you provided in this study? What encouraged you to participate in this study?*
  - If no, *why not?*
- *What situations or settings would encourage you to donate your medical data?*
- *What kind of information about data donation is important for you when you make your decision?*
- *Is it necessary for you to be able to choose which data to donate and which medical studies this data can be used in? Why?*
- Let's consider a specific scenario: There is an app that users can use to donate their medical data to research institutes. The collected data will be used by researchers to better understand diseases and improve public health. The Corona-Datenspende-App (Corona data donation app) by the Robert Koch institute (RKI) and the SafeVac app by the Paul Ehrlich institute are two examples of such app. In the Corona data donation app, the donated data is collected from users' fitness trackers like an Apple watch. In the SafeVac app, the donated data is collected via a questionnaire. *Have you heard about any of these two apps?*
- Drawing Exercise: *Can you please draw on these papers how you think a medical data donation app like the Corona data donation app or the SafeVac app works, including the data flow?* When you draw, please keep in mind how things work in this app behind the scenes. Also, please think aloud while you draw your sketch so that I can understand what you are drawing and why you are drawing it. Another important remark: keep in mind that there are no correct answers to the questions—just answer them based on your own knowledge and experiences.
- Considering the infrastructure that you have just drawn: *what kind of data can the medical data donation app know about the data donor? Where is the data stored?*
- Now, use different colors to mark *which entity stores data about you, and which entity has data that can be linked to your person. Please also list the specific data that is stored, for instance, medical data, and demographic data.* Be as specific as possible.
- Please mark *what information should be protected about the data donors? From what should the data donors be protected?*
- *Do you have an opinion regarding who is responsible for providing this protection?*
- Depending on what the participant drew: *Which medical or personal information do you consider so sensitive that you would refuse to share it with a medical data donation app?*
- According to your understanding, *who can access your donated data?*

- *What information about the data donors can the researchers obtain?*
- *Would you still agree to donate data if you knew the medical data donation app could link your donated data to your name, mobile phone number, or location?*
  - *If yes, why?*
  - *If not, will you change your mind if you know the application is run by an official authority? Why?*
- *Have you heard about anonymity? What does it mean to you in the context of medical data donation?*
- *How, to your understanding, does a medical data collection app like the Corona data donation app or the SafeVac app protect the anonymity of the data donors?*
- *Which entity or entities are responsible for ensuring anonymity?*
- *What does breaking the anonymity of data donors mean, in your opinion?*
- *How, in your opinion, can the anonymity of data donors be broken?*
- *Is it still possible to compromise the anonymity of data donors if the donated data does not contain information that explicitly identifies an individual, such as a name, social security number, phone number, address, or driver's license? Why?*

ERKLÄRUNG

Hiermit erkläre ich, die vorgelegte Arbeit zur Erlangung des akademischen Grades Doktor rerum naturalium (Dr. rer. nat.) mit dem Titel

Improving Anonymity in Public Group Communication Scenarios

selbständig und ausschließlich unter Verwendung der angegebenen Hilfsmittel erstellt zu haben. Ich habe bisher noch keinen Promotionsversuch unternommen.

Bochum, 16.10.2024

Sarah Abdelwahab Kamel Fayed Gaballah